Estimation de mouvement et segmentationPartie I : Estimation de mouvement par ondelettes spatio-temporelles adaptées au mouvement.Partie II : Segmentation et estimation de mouvement par modèles de Markov cachés et approche bayésienne dans les domaines direct et ondelette.

Patrice Brault

▶ **To cite this version:**

Patrice Brault. Estimation de mouvement et segmentationPartie I : Estimation de mouvement par ondelettes spatio-temporelles adaptées au mouvement.Partie II : Segmentation et estimation de mouvement par modèles de Markov cachés et approche bayésienne dans les domaines direct et ondelette.. Traitement du signal et de l'image [eess.SP]. Université Paris Sud - Paris XI, 2005. Français. NNT : . tel-00011310

HAL Id: tel-00011310
https://theses.hal.science/tel-00011310

Submitted on 6 Jan 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

N^o D'ORDRE

UNIVERSITE PARIS-SUD XI
Faculté des Sciences d'Orsay

THÈSE DE DOCTORAT

SPECIALITE : PHYSIQUE

Ecole Doctorale "Sciences et Technologies de l'Information des Télécommunications et des Systèmes"

Présentée par :

Patrice BRAULT

Thèse préparée dans les laboratoires :

IEF, Institut d'Electronique Fondamentale, CNRS UMR-8622.
LSS, Laboratoire des Signaux et Systèmes, CNRS UMR-8506.

Sujet :

Estimation de mouvement et segmentation d'image

Partie I : Estimation de mouvement par ondelettes spatio-temporelles adaptées au mouvement

Partie II : Segmentation et estimation de mouvement par modèles de Markov cachés et approche bayésienne dans les domaines direct et ondelette.

(Version préliminaire, approuvée pour soutenance et reproduction)

Soutenue le 29 novembre 2005 devant les membres du jury :

Jean-Pierre ANTOINE	Rapporteur	Professeur, Université catholique de Louvain
Xavier DESCOMBES	Rapporteur	Chargé de Recherches INRIA, HDR, Sophia-Antipolis
Pierre DUHAMEL	Président	Directeur de Recherches CNRS, LSS Supélec
Alain MERIGOT	Directeur de thèse	Professeur, IEF, UniversitéParis-sud Orsay
Ali MOHAMMAD-DJAFARI	Directeur de thèse	Directeur de Recherches CNRS, LSS Supélec
Jean-Luc STARCK	Examinateur	Ingénieur Chercheur, HDR, CEA Saclay

A Estelle,
à mes “Papus”,
à mes parents,
André et Marie-Paule.

La vérité que je révère,
c'est la modeste vérité de la science,
la vérité relative, fragmentaire, provisoire,
toujours sujette à retouche, à correction,
à repentir, la vérité à notre échelle ;
car tout au contraire, je redoute
et je hais la vérité absolue,
la vérité totale et définitive,
la vérité avec un grand V,
qui est la base de tous les sectarismes,
de tous les fanatismes et de tous les crimes.

Jean Rostand
Le droit d'être naturaliste, 1963.

Remerciements

J'adresse ces tout premiers remerciements à Monsieur Jean-Michel Lourtioz, directeur de l'Institut d'Electronique Fondamentale, qui a pris le temps d'examiner avec attention ma démarche de préparation d'une thèse et qui m'a donné les moyens de m'y engager, tout d'abord à mi-temps avec mon travail d'ingénieur, puis à plein temps pour la deuxième moitié de cette thèse. Il a su reconnaître ma motivation dans ce travail et je le remercie très vivement de m'avoir encouragé dans ma démarche et de m'avoir fourni les moyens de rendre possible l'aboutissement de cette thèse.

Mes remerciements s'adressent ensuite à Alain Mérigot, mon directeur de thèse, vers qui j'ai cru bon de me tourner pour cette direction de thèse car j'ai senti en lui le profil d'un passionné de recherche. Qu'il soit remercié pour avoir volontiers accepté cette tâche d'encadrement.
Je tiens à remercier aussi les membres du département Axis et en particulier les professeurs Roger Reynaud et Bertrand Zavidovique, qui, en début de thèse, m'ont non seulement permis de m'engager dans cette voie mais m'y ont encouragé alors que je voyais difficilement réalisable cette entreprise. Je crois que cet élan dans le sens d'un fervent dévouement pour l'enseignement et la recherche est tout à leur honneur.

Je me tourne maintenant vers Ali Mohammad-Djafari qui a accepté avec une grand gentillesse et une grande patience d'encadrer la deuxième partie de mon travail de thèse au sein du GPI. Il a accepté de former aux problèmes inverses et aux méthodes bayésiennes un thésard qui n'avait sans doute pas tout le bagage requis pour appréhender cette tâche avec facilité. Je le remercie ici chaleureusement pour sa disponibilité, sa passion et son dévouement dans cet encadrement.

J'ai eu la chance d'avoir les approbations de Jean-Pierre Antoine et de Xavier Descombes pour être rapporteurs de cette thèse. C'est avec beaucoup de sympathie que je les remercie d'avoir accepté la lourde tâche de relecture et de critique de cette thèse. Leur lecture attentive et leurs très intéressantes remarques m'ont permis de corriger nombre de points et d'améliorer la qualité de cette thèse.

Je remercie vivement Pierre Duhamel pour avoir m'avoir fait l'honneur d'accepter la présidence de mon jury et Jean-Luc Starck pour m'avoir fait celui d'examiner cette thèse.
J'adresse aussi d'exceptionnels remerciements à Michel Fliess et au professeur Fionn Murtagh pour avoir eu tous deux l'extrême gentillesse de se proposer pour mon jury de thèse. Nos règles administratives n'ont, hélas, pas pu me permettre d'accepter leurs très sympathiques propositions.

D'autre part je tiens à remercier sincèrement toute l'équipe du GPI/LSS pour m'avoir accueilli dans son groupe et pour avoir montré beaucoup de disponibilité et de passion pour répondre à mes questions. Je remercie et salue en particulier le Capitaine Guy Demoment, qui, lorsque tout "abafointé", sait prendre avec une grande sagesse le parti de s'en f....., Jean-François Giovannelli "Gio l'Americag...", Thomas, Adel "maxi persil", Olivier, Mehdi "Orion", Aurelien "ChessMaster", Fabrice, Zouaoui, Nadia et tous ceux qui ont récemment intégré le GPI. Je souhaite aux thésards de profiter pleinement de l'encadrement très enrichissant dont ils ont la chance inestimable de pouvoir bénéficier ainsi que des excellents cours de Guy et d'Ali.

Je salue et remercie ici le professeur Goutelard qui a su m'encourager dans la voie de la recherche et qui a guidé mes premiers pas dans cette voie.

Un grand merci à tous mes amis et collègues qui m'ont encouragé et aidé, et en particulier à Hugues, Annick et Mohammed sans lesquels ce travail n'existerait peut-être pas. Mes excuses les plus "musicales" à mes nouveaux amis de l'ensemble Ebène Bleu, et en particulier à Pierre-François, que j'ai lâchement abandonnés en ce début d'année de soutenance. Je leur suis très reconnaissant de leur compréhension et de leur marque de sympathie.

J'adresse un regard complice vers ma tendre compagne qui m'a toujours encouragé tout au long de ce travail et vers mes trois "Papus" qui n'ont pas manqué de me demander régulièrement, alors que j'attirais leur attention vers une bouteille de lait, : "Papa, c'est quoi un problème inverse ?".

Enfin je dédie cette thèse à mes parents, André et Marie-Paule, qui ont su me rappeler de bons souvenirs aux moments difficiles et ont su aussi me rappeler l'importance de ma tâche.

Table des matières

Introduction générale

Cette thèse concerne l'analyse de séquences d'images et plus particulièrement l'estimation de mouvement et la segmentation. Le but premier de ce travail était d'aborder la question de la compression vidéo par des méthodes à la fois originales, nouvelles et mettant en jeu une vision de la compression plus contextuelle que celle qui est couramment développée, dans les normes récentes, par "comparaison de blocs".

Elle a été développée en deux parties : une première partie sur l'estimation de mouvement fondée sur des familles d'ondelettes construites pour l'analyse et la quantification du mouvement et une deuxième partie sur la segmentation par approche bayésienne et champs cachés de Gauss-Markov.

La première partie a été initialement inspirée par la norme MPEG4 dans laquelle la compression se fait en partie par une approche objet. Le terme "orienté objet" dans la norme MPEG4 consiste principalement à segmenter les régions d'intérêt (ROI) pour les coder ensuite avec une plus ou moins bonne qualité (donc un taux de compression plus ou moins élevé) selon l'intérêt que l'on porte à cette région. Cette première fonctionnalité vise à compresser le flux vidéo par adaptation de la qualité aux régions d'intérêt de la scène. Cependant dans l'approche MPEG4, le codage spatial d'un objet d'intérêt reste un codage par blocs et l'estimation/compensation de mouvement est aussi réalisée de façon brute par des vecteurs de mouvement représentant simplement les translations des blocs. Il nous a paru intéressant, dans le cadre d'une approche objet, de chercher plutôt à caractériser les mouvements des objets segmentés d'une séquence et à chercher un moyen de faire une prédiction de mouvement fondée sur la trajectoire des objets plutôt que de rester sur l'estimation brute du mouvement des blocs constituant ces objets. En effet l'inconvénient des méthodes "génériques" basées sur un découpage en blocs est qu'elles ne s'intéressent pas au contenu des séquences.

L'estimation de mouvement prise dans un sens "contextuel" sur des groupes d'image (GOP ,Group of Pictures ou GOF, Group of Frames), apporte des informations plus précises qui permettent de réaliser des opérations telles que l'analyse du mouvement sur des objets, ou régions, d'une scène ainsi que la segmentation de ces objets. Elle peut conduire aussi à l'identification de trajectoires des objets ainsi qu'à l'analyse de scène. C'est cette approche contextuelle des scènes que nous avons mise en avant car elle semble indissociable, à terme, de méthodes de compression plus complexes mais "intelligentes" et performantes.

Afin de quantifier les paramètres du mouvement des objets dans une scène, nous nous sommes donc intéressés plus particulièrement à des techniques de filtrage et d'analyse par ondelettes spatio-temporelles. Nous nous sommes orientés vers des familles d'ondelettes redondantes, développées pour l'analyse du mouvement, et qui ont déjà été utilisées dans des applications de suivi de cibles. Cependant les trajectoires des objets dans de telles applications restent relativement simples. L'utilisation dans des séquences vidéo présente la difficulté de quantifier des paramètres de mouvements quelquefois beaucoup plus complexes. L'utilisation de ces familles (ou frames) adaptées au mouvement nécessite alors d'utiliser un dictionnaire assez large d'ondelettes adaptées qui permet d'extraire les paramètres comme la vitesse, la rotation ou la déformation, paramètres qui ne peuvent être obtenus par des approches par blocs.

Dans la deuxième partie nous abordons le problème de la classification/segmentation pour des images fixes et pour des séquences d'images dans le domaine direct, par une approche bayésienne. Nous présentons aussi une approche de l'estimation du mouvement et de la compression en nous basant sur les résultats obtenus dans la segmentation statique. Une approche "itérative" de la segmentation bayésienne réalisée sur des séquences d'images permet d'améliorer la vitesse de segmentation. A partir de la segmentation de séquences nous présentons une méthode d'estimation de mouvement basée sur la classe d'appartenance des objets et sur leur "masse" (nombre de pixels). Le suivi d'un objet dans une scène est alors rendu possible ainsi que l'estimation de son déplacement, en se basant sur le mouvement de son centre de masse. Nous présentons finalement le développement d'une méthode de segmentation bayésienne et champs de Markov cachés dans le domaine transformé des ondelettes. Les résultats obtenus par projection dans le domaine transformé des ondelettes montrent que la segmentation peut se faire beaucoup plus rapidement que dans l'espace direct.

Les deux parties de cette thèse peuvent être vues comme deux approches réciproques de l'estimation de mouvement et de la segmentation dans le but de réaliser l'analyse et la compression orientée objet de séquences vidéo. En effet, dans la première partie une séquence vidéo est analysée en termes de paramètres de mouvement. Le but est de suivre des groupes de pixels possédant des paramètres semblables. Ces pixels appartiennent à des régions homogènes au sens du mouvement et permettent donc d'extraire, ou de segmenter, des régions ou des objets distincts. Dans la seconde partie nous effectuons d'abord une segmentation de la séquence basée sur l'homogénéité de niveaux de gris. Le mouvement des régions ou objets segmentés est paramétré en se fondant sur le mouvement du centre de masse des régions. Dans les deux cas le but est de permettre l'identification de la trajectoire d'une région plutôt que la valeur d'un simple vecteur de mouvement entre deux blocs de l'image. En ce sens nous pensons que les deux approches présentées dans la suite conviennent à une approche "contextuelle" et intelligente de la compression.

Partie I : Estimation de mouvement par ondelettes adaptées au mouvement

Introduction à la première partie

Cette première partie sur l'estimation de mouvement a débuté par une réflexion sur la norme de compression MPEG4 orientée objet. Cette norme, dans son état de développement en 1999, était prévue pour réaliser la compression vidéo en prenant en compte, et c'était une différence notoire avec MPEG2, la *notion contextuelle* des scènes. Plus précisément, la norme MPEG4 prend en compte les objets, ou régions d'intérêt (ROI), de façon individuelle dans une séquence, en effectuant une segmentation et en réalisant une compression à taux variable en fonction de caractéristiques particulières de ces objets (vitesse, forme, couleur). Cela représente une innovation importante par rapport aux méthodes de codage. Cependant il semblait insuffisant de coder de la même manière, aussi basique, c'est-à-dire par comparaison de blocs (BM, Block-Matching), des objets segmentés et les régions statiques ou "sprite" qui entouraient ces objets dans l'image. Segmenter un objet puis faire de l'estimation de mouvement par blocs sur ce même objet semblait un peu limitatif. Nous nous sommes donc intéressés à l'aspect estimation de mouvement par objet plutôt que par blocs car le développement de méthodes orientées objet nous semblait devoir nécessairement comporter tôt ou tard un aspect beaucoup plus "contextuel" qui est *celui du déplacement de l'objet* lui-même. C'est sur la base de l'estimation du déplacement des objets, et non plus des blocs qui les représentent, que l'approche objet nous paraissait devoir progresser.

L'objectif de la première partie de cette thèse est d'investiguer et d'utiliser les capacités de familles d'ondelettes redondantes pour l'estimation du mouvement dans des séquences vidéo ainsi que pour l'analyse du mouvement (lire par exemple [BH95, Tru98, JMR00] pour une introduction générale aux ondelettes). L'utilisation pour la compression non pas systématique, mais contextuelle, est mise en avant dans ce travail. Par contextuelle, nous entendons une compression adaptée au contenu de la scène et aux besoins de l'application. En particulier, et comme cela est déjà défini dans MPEG4, on peut adopter un faible taux de compression pour toute partie d'une scène qui présente des critères d'intérêt essentiels (objets de taille, de couleur, de mouvement particuliers par exemple). Une analyse de la scène fournissant une quantification plus précise des mouvements devrait permettre de réaliser encore mieux une compression contextuelle. C'est ce type d'analyse qui a motivée notre approche.
Les familles d'ondelettes redondantes [0] ont fait l'objet d'un intérêt relativement restreint dans les

[0] La transformée en ondelettes présente l'information initialement contenue dans un espace de dimension n, le signal, dans un espace de dimension $n+1$, le plan temps-échelle. Elle met donc en oeuvre, pour représenter l'information, un nombre de coefficients beaucoup plus important que nécessaire : c'est une représentation redondante. Une même fraction de l'information y est reproduite plusieurs fois, partagée par différents coefficients de cette représentation

années passées car elles sont coûteuses en puissance de calcul. Les algorithmes ont un ordre de complexité, pour des signaux 1D, en $\mathcal{O}(kN^2log(N))$ alors qu'il est en kN^2, où k est la longueur de la séquence du filtre, pour les décompositions non-redondantes (algorithme orthogonal pyramidal de S. Mallat).
Les ondelettes redondantes sont un outil d'analyse [MZ92, Tor95, AAE+95, Abr97, AMVA04] et de restauration d'images [SMB00] alors que les ondelettes non-redondantes sont utilisées essentiellement pour la compression et le débruitage [Ber99b, LP02]. La décomposition redondante dyadique de Mallat a été utilisée dans la décomposition-reconstruction d'une image en n'utilisant presque exclusivement que les contours détectés par des ondelettes dérivées d'une gaussienne. L'algorithme à trous développé par M. Holschneider, R. Kronland-Martinet, J. Morlet et P.Tchamitchian dans [GKMM89] puis par M. Shensa [She92], permet de réaliser une décomposition de l'image sans réduction de sa résolution. Les applications en restauration fondées sur cette décomposition redondante sont nombreuses notamment en astronomie (Starck, Murtagh, Bijaoui dans [SMB00]) et très récemment en restauration vidéo. Le développement des "ridgelets" et des "curvelets" basées sur ce même algorithme à trous ont permis d'atteindre une qualité très élevée en restauration (E. Candes, D. Donoho, J.L. Starck dans [SCD01] .

D'autre part la compression à base d'ondelettes orthogonales a connu deux développements récents : les "bandelettes" (bandlets) (E. Le Pennec, S. Mallat [LPM00, LP02, LPM03]) pour la compression d'images statiques et le calcul du flot optique rapide par projection sur des bases orthogonales pour l'estimation/compensation de mouvement (thèse de C. Bernard [Ber99b] avec S. Mallat).

D'autres utilisations des ondelettes pour la compression et l'estimation de mouvement ont été réalisées par l'IRISA [VG03] pour le filtrage de trajectoires d'objets (ondelettes simples de Haar) ou encore par schéma "lifting" (cf. annexe) réduit à l'étape de prédiction (M. Barret []).

En revanche, les familles d'ondelettes spatio-temporelles développées pour l'analyse de mouvements affines (rotation), pour la déformation (transformation quelconque [Combes]) ou pour la détection cinématique (vitesse , accélération) ont connu peu d'intérêt. Ces familles sont un développement de la classique décomposition en ondelettes qui inclut la translation et la mise à l'échelle spatiales. Elles prennent tout d'abord en compte les mêmes transformations dans le domaine (spatial + temporel) et permettent la détection et la quantification du mouvement. Nous avons abordé ces groupes particuliers de familles d'ondelettes étudiées notamment par (Duval-Destin, Murenzi, Dutilleux puis J.P. Leduc et al. dans [DDM93, Dut89, LMMS00]) et avons imaginé tout d'abord leur utilisation dans un schéma d'estimation de mouvement en développant un nouvel algorithme plus rapide basé sur l'analyse multi-résolution à trous. Le développement de cet algorithme présente la difficulté d'adapter les ondelettes spatio-temporelles à des paramètres cinématiques tout en conservant la propriété d'une analyse multirésolution (relation de double échelle) indispensable à l'algorithme à

[Abr97], §1.1.3. S. Mallat, dans sa transformée pyramidale orthogonale, réduit totalement cette redondance en créant une orthogonalité complète entre TOUS les coefficients (et les sous-espaces auxquels ils appartiennent), intra et inter-échelles. La transformée redondante est donc "prolixe" et peu efficace pour la compression, mais s'avère beaucoup plus intéressante pour suivre avec une grande finesse l'évolution d'un signal. Ce sont donc ses *capacités d'analyse* qui nous intéressent, dans cette approche d'EM, bien que celle-ci vise, c'est paradoxal, la compression vidéo.

trous. Cette difficulté n'a pas été perçue immédiatement et les premiers résultats de cette adaptation n'ont pas apporté de résultats suffisamment probants en ce qui concerne la qualité de l'analyse. Nous avons alors commencé l'étude de l'adaptation d'ondelettes Splines au mouvement. Cette approche présente le triple intérêt d'utiliser des ondelettes compactes dans les domaines direct et dual, de travailler avec des ondelettes peu oscillantes qui sont plus adaptées que les ondelettes de Morlet à l'analyse d'objets aux bords bien définis et enfin d'être déjà utilisées avec l'algorithme à trous en restauration [SCD01].

Un deuxième volet à l'approche par mesure de paramètres cinématiques par ondelettes spatio-temporelles a été d'imaginer la construction des trajectoires des objets dans une scène vidéo. La construction de trajectoire est alors vue comme une extension de la mesure, basique, du mouvement par vecteurs représentant le déplacement linéaire de blocs, ou d'objets, dans des scènes. Le but est d'associer, à un bloc de l'image ou à un objet (région homogène), une trajectoire qui serait construite sur les toutes premières trames d'une scène. La construction de cette trajectoire spatio-temporelle nécessite de connaître les paramètres cinématiques de l'objet. Ceux-ci sont obtenus soit par mesure de champ dense (flot optique), soit par mesure à partir des positions prises par l'objet dans les trames successives (cadencées par la période vidéo), soit à partir d'ondelettes spatio-temporelles (ST) adaptées au mouvement qui déterminent dans une scène les objets présentant une cinématique déterminée. Nous verrons que cette estimation se fait à partir de plusieurs ondelettes mères au départ afin d'obtenir une cartographie relativement complète des caractéristiques cinématiques des objets. Puis [Mujica, Leduc...], un objet particulier peut être "suivi" dans la scène. L'analyse et la compression peuvent alors aller de pair : comme pour les objets en mouvement rapide sur fond lentement variable (séquence de tennis "Edberg"), nous pouvons nous intéresser à la compression de toute une scène avec un taux de compression élevé tout en ne conservant que le ou les objets dont la cinématique nous intéresse.

La première partie de ce manuscrit est organisée de la façon suivante. Hormis cette introduction, le premier chapitre présente un bref état de l'art en compression statique d'image, ou compression spatiale, et introduit naturellement les techniques de compression vidéo fondées sur la compression spatiale et la compression temporelle. Les deux grandes familles de codeurs, hybrides et 3D, y sont présentées. Nous y présentons les approches de codage avec et sans compensation de mouvement, ainsi que les codeurs sans transmission du mouvement. Le chapitre est complété par un tour d'horizon des normes de compression vidéo. Le deuxième chapitre présente un état de l'art des méthodes d'estimation du mouvement : estimation par comparaison de blocs (BM, block-matching), filtrage spatio-temporel, flot optique, non-transmission des vecteurs de mouvement. Le chapitre 3 est une introduction aux techniques spectrales d'estimation du mouvement par la transformée de Fourier. Il établit le lien entre un déplacement dans le domaine spatio-temporel direct et sa résultante dans le domaine spectral. Il rappelle les possibilités d'analyse du mouvement qu'offre le domaine spectral et nous amène naturellement au chapitre 4. Dans celui-ci l'analyse spectrale est réalisée non plus sur la base des fonctions continues ou à fenêtre, qu'offrent les transformées de Fourier, de Fourier à fenêtre (STFT, short term Fourier transform) ou de Gabor, mais sur celle d'une famille de transformées en ondelettes adaptées au mouvement (MTSTWT, Motion tuned spatio-temporal wavelet transform). Le chapitre 5 nous montre quels sont les résultats obtenus en appliquant ce type de transforma-

tion à une séquence vidéo et fait la comparaison avec une méthode récente de résolution rapide du flot optique développée par C. Bernard et S. Mallat. Enfin le chapitre 6 présente un schéma d'estimation de mouvement, basé sur la MTSTWT, et dans lequel nous proposons le fusionnement de deux méthodes : l'estimation des paramètres de mouvement d'objets et l'identification rapide de leur trajectoire. Cette approche "contextuelle" est proposée dans le cadre de futures méthodes de compression "intelligentes" de séquences dans lesquelles on ne base plus seulement la réduction de redondance temporelle sur la connaissance sommaire de vecteurs de mouvement mais sur la connaissance de la trajectoire actuelle des objets dans une scène. Cette approche "contextuelle" de la compression est donc basée sur l'analyse de la scène, que nous pensons être une des voies potentielles pour réaliser, à terme, une compression efficace d'une séquence.

Chapitre 1

La compression vidéo

La compression vidéo peut s'entendre de plusieurs façons : compression "en-ligne" avec latence faible ou moyenne (compression "temps réel" avec latence de l'ordre de quelques trames de 20ms), avec latence forte (ordre de la seconde), ou encore "hors-ligne" avec latence quelconque. De même que pour la compression d'images fixes, la qualité de restitution en terme de rapport signal/bruit joue un rôle prédominant. Nous nous sommes intéressés tout d'abord à l'intégration de l'estimation de mouvement dans des normes de codage classiques notamment la norme MPEG4 qui présentait au moment où cette thèse a commencé l'intérêt d'une mise en oeuvre de la compression dans une orientation "objet". Nombre de difficultés techniques ont retardé la mise en application de cette norme dont le cahier des charges, un peu exigeant et complexe, s'est avéré trop lourd pour les applications industrielles attendues. La "compression orientée objet" qui semblait faire tout l'intérêt de cette norme a montré très tôt qu'elle ne se résumait, en ce qui concerne le codage du mouvement, qu'à une estimation par déplacement de blocs très (trop) classique et non par déplacement d'objets ou de régions caractéristiques. C'est dans ce sens que nous avons choisi de tester d'autres approches plus précises quant à la quantification du mouvement par régions, notamment avec des ondelettes sensibles au mouvement.

Dans ce chapitre nous présentons les familles les plus répandues de codeurs vidéo ainsi que les deux grandes techniques utilisées plus particulièrement sur les codeurs hybrides ou en général sur les codeurs avec compensation de mouvement puisque c'est l'estimation du mouvement qui nous intéresse ici. Nous ferons bien sûr une part importante à l'analyse de scène qui intervient de plus en plus dans des codeurs dont la complexité est grandissante et dont la gestion de l'aspect "contextuel" renforce l'efficacité de la compression. En effet la compression peut être vue de façon tout à fait non-supervisée ou "brute", mais aussi de manière plus orientée vers des caractéristiques particulières de la scène à coder : compression plus forte de l'arrière-plan ("sprite") quasi-immobile, ou compression forte des régions de grande dimension, de couleur particulière, etc.

1.1 La compression statique

La compression vidéo ne peut être présentée sans une brève introduction sur la compression statique. Nous rappelons les grandes lignes des normes JPEG, JPEG2000 et des algorithmes de type SPIHT. Qu'attend-on d'une compression : une réduction de la taille des données, donc du débit binaire lors d'une transmission continue. Celle-ci est dépendante de trois paramètres principaux : la vitesse de codage et décodage, le taux de compression, la qualité de compression. La complexité du codeur et du décodeur est un des éléments importants. A cela s'ajoutent des caractéristiques comme l'échelonnabilité. Il s'agit de pouvoir tronquer une image déjà codée afin de la transmettre à une moins bonne résolution sans devoir envoyer un code complet puis décoder ce code et n'en garder qu'une partie conduisant à une qualité réduite.

1.2 La compression vidéo

1.2.1 Redondance spatiale

La compression spatiale consiste en une réduction de la redondance spatiale de l'information. Celle-ci est souvent fondée sur le codage de l'image dans un domaine transformé permettant une représentation "creuse" du signal ou de l'image. Originellement les transformées les plus utilisées sont : la DCT (Discrete Cosine Transform), l'OWT (Orthogonal wavelet transform), l'ACP (analyse en composantes principales, ou PCA, ou encore SVD ou KLT). Ces transformées sont utilisées pour la compression statique aussi bien que vidéo. La DCT utilisée en MPEG2, est opérée par bloc de 8x8 ou 16x16 pixels suivie d'une quantification et d'un codage entropique (table de Huffman) qui attribue des symboles courts aux valeurs d'échantillons les plus souvent rencontrées.
D'autres méthodes de codage spatial sont basées sur des approches échelonnables (quantification progressive), hiérarchiques ou par plans de bits : c'est le cas des codeurs EZW (Embedded Zero-Tree Wavelet) [Sha93], SPIHT (Single Partition Hierarchical Tree) [SP96] et EBCOT (Embedded Coding With Optimized Truncation) de JPEG2000 (Taubman, [Tau00, TZ94]).
Dans le schéma EZW, on cherche à encoder les coefficients d'ondelette de la façon la plus compacte possible. Sachant qu'un coefficient d'intérêt, donc de forte valeur, est souvent suivi par des coefficients "enfants" (de l'échelle inférieure donc de la résolution supérieure) dont la probabilité est forte d'être aussi de forte valeur, la manière la plus rapide pour coder les coefficients d'intérêt est de suivre un arbre de codage à partir de chaque coefficient de forte valeur depuis la plus basse résolution. Dès qu'apparaît dans un arbre un coefficient faible ou nul, ses "enfants" ont aussi une forte probabilité d'être nuls et l'on encode le reste de l'arbre par un symbole représentant une suite de zéros (zero-tree). Le décodage est donc aussi extrêmement simplifié dès qu'apparaît le symbole d'un arbre de zéros.
Le schéma SPIHT de Said et Pearlman [SP96], améliore l'algorithme EZW en partionnant l'espace des coefficients selon une échelle de décroissance de leur amplitude par octave, effectue la transmission par plans de bits (échelonnabilité) et utilise l'auto-similarité des coefficients inter-échelles.

1.2.2 Redondance temporelle

La redondance temporelle consiste à représenter l'image "n" de la séquence à partir de l'image "n-1" en transmettant uniquement la valeur des vecteurs de mouvement des blocs entre deux trames (ou images) de la séquence. La redondance provient du fait que si l'éclairement de la scène est à peu près constant et qu' aucun objet n'apparaît ou disparaît, chaque bloc de l'image "n-1" doit se retrouver dans l'image "n" et donc le codage complet du bloc "n" devient inutile : il suffit alors d'associer à un bloc un simple vecteur de mouvement, codé par quatre coordonnées (x_{n-1}, y_{n-1}) et (x_n, y_n) dans le plan image, pour replacer le même bloc dans l'image "n", sans en refaire le codage spatial complet.
La prédiction temporelle peut, dans plusieurs schémas et normes classiques (MPEG, H26), être associée à la notion de groupe de trames, les GOF (group of frames), ou GOP (group of pictures). Dans MPEG-1 et -2 les trames peuvent être codées de trois façons : I pour Intra, P pour Prédite et B pour Bidirectionnelle. Le codage Intra d'une image consiste en un codage spatial intégral de l'image (par exemple la DCT ou transformée discrète en cosinus de MPEG2), comme s'il s'agissait d'une image fixe. Le codage Intra ne fait donc pas appel à la notion de redondance temporelle et ne réalise pas d'estimation/prédiction du mouvement des blocs.
Dans le codage d'une trame "P", on s'intéresse à la ressemblance, ou la distance, entre des blocs de l'image actuelle (n) et des blocs de l'image passée (n-1). De façon sommaire, on peut dire que si un bloc de la trame actuelle (n) présente un niveau moyen de gris similaire à un bloc de la trame (n-1) et se trouve dans un voisinage spatial défini du bloc (n), on dit qu'il y a redondance temporelle du bloc. Plus précisément, le "niveau de gris moyen" est donné par l'EQM (erreur quadratique moyenne) mais plus souvent par l'EAM (erreur absolue moyenne). En ce qui concerne le voisinage spatial, ou "zone de recherche", elle est centrée sur la position du bloc initial et ne s'étend pas au-delà des "composantes maximales" d'un vecteur mouvement. Afin d'améliorer la recherche du bloc le plus semblable dans une zone, plusieurs techniques existent. Elles consistent par exemple à ne calculer un critère de ressemblance que pour les blocs ne se recouvrant pas ou encore à commencer la recherche par le centre de la zone, puis à tourner autour en cherchant toujours à augmenter la ressemblance. Nous faisons ici référence à [Gui96], section 4.4, pour des explications plus détaillées sur ces aspects de mise en correspondance.

Nous venons donc de montrer qu'il n'est pas nécessaire de recoder spatialement le même bloc dans la trame (n). Il suffit de réutiliser le codage spatial du bloc (n-1) et de lui adjoindre un vecteur de mouvement $V\{(i_1, j_1), (i_2, j_2)\}$ correspondant à son déplacement entre les trames (n-1) et (n). Dans ce cas on ne transmet plus, pour ce bloc de la trame (n), que le vecteur de mouvement correspondant. Ce système nous amène à la notion de prédiction temporelle qui trouve son explication si l'on fait l'hypothèse que le même bloc aura, entre les trames (n) et (n+1), subi un vecteur de déplacement de même amplitude et orientation qu'entre les trames (n-1) et (n). Dans ce cas, sans chercher à trouver la position du bloc dans la trame (n+1), on lui attribue une position par "prédiction" d'un déplacement similaire à celui qu'il a connu entre les trames (n-1) et (n).
Enfin le codage bidirectionnel d'une trame fait appel aux deux types de codage : I et P. Les images bidirectionnelles sont codées, comme leur nom l'indique, à partir des trames codées "I" et "P" et sont placées entre celles-ci. Le schéma 1.1 donne un exemple de la façon dont sont calculés les trois

types de trames. Selon le taux d'erreur, la vitesse de transmission ou la qualité que l'on désire obtenir, le schéma du "GOF" variera et pourra être de type IBBP comme celui indiqué dans la figure ou bien IPPIPP, ou l'on cherche à supprimer le nombre de trames bidirectionnelles coûteuses en calcul.

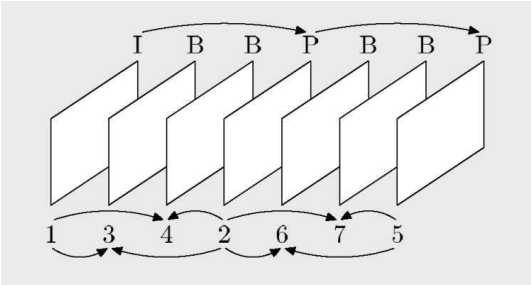


FIG. *1.1 – Trois types de codage de trames : Intra, Prédite et Bidirectionnelle et l'entrelacement de ces codages dans une séquence "GOF"*

1.2.3 Premiers schémas de compression vidéo par ondelettes

Les deux schémas qui suivent relèvent de la compression spatiale bien qu'ils fassent intervenir les différences entre images (pour le schéma DPCM).

M-JPEG

Le premier schéma de codage vidéo est réalisé de façon assez naturelle par le codage/décodage statique et individuel de chaque trame. Un codage de type JPEG pour chaque trame a donné le schéma "Motion-JPEG" (Fig. 1.2) dans lequel seule la redondance spatiale est exploitée dans chaque trame prise séparément. La redondance temporelle n'est pas exploitée. Un autre type de compression spatiale pourrait être adopté. Cependant le codage JPEG offre la possibilité d'une qualité (et donc d'un taux de compression) variable et élevé (avec perte).

JPEG-DPCM

Le deuxième schéma utilisant la modulation différentielle des images par impulsions codées (JPEG-DPCM, Differential Pulse Coded), voir Fig. 1.3, commence à exploiter cette redondance temporelle. Il s'agit alors simplement de coder et de transmettre la différence, codée en JPEG, qui existe entre deux images successives. Seule la première image de la séquence est codée en JPEG et transmise totalement. L'erreur différentielle a bien sûr tendance à se propager dans un schéma non-bouclé, cependant elle reste faible car non-prédictive, mais parfaitement déterministe.

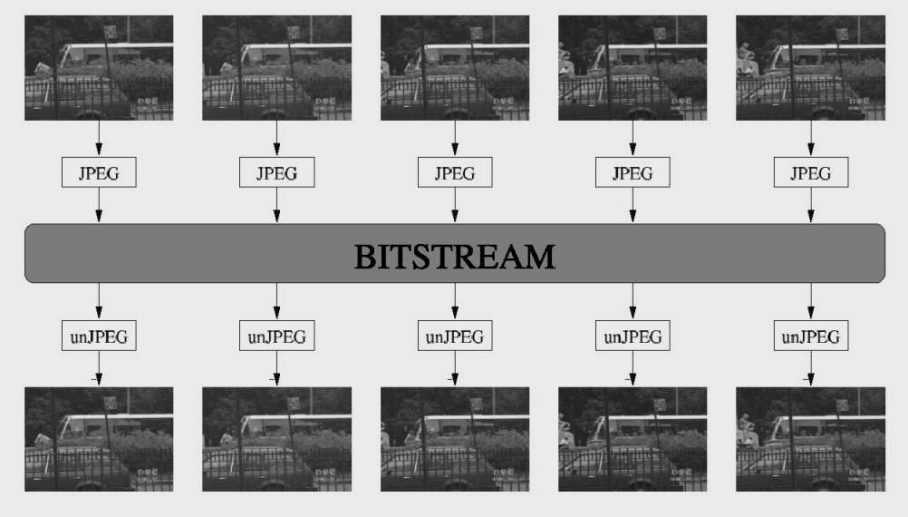


FIG. *1.2 – Codeur/décodeur M-JPEG (Motion-JPEG)*

1.2.4 Codeurs hybrides

La compression vidéo, au niveau de la couche physique, est basée, dans les premières normes, sur un "codage hybride". Cette appellation trouve sa signification dans le fait que la compression se fait par une méthode spatiale et une méthode temporelle. La compression consiste en une étape de suppression de la redondance spatiale (ou décorrélation spatiale) et une étape de suppression de la redondance temporelle. Cette dernière consiste à éviter de transmettre, pour deux images successives, les blocs des deux images qui présentent la même "information moyenne", c'est-à-dire le même niveau de luminance moyen. Ainsi si l'on peut retrouver, dans une image n, un bloc Bi_n de même caractéristique que le bloc Bi_{n-1}, alors il suffit de ne transmettre que le vecteur de mouvement correspondant au déplacement du bloc Bi_n pour transmettre la deuxième image. La mesure du déplacement est réalisée entre deux *images successives* de la séquence dans le cas des normes telles que MPEG2, MPEG4.
On recherche la "meilleure correspondance" entre deux blocs de pixels 16×16 situés dans des trames consécutives. Cette correspondance peut se faire par minimisation de l'erreur quadratique entre les illuminations correspondant aux pixels respectifs des deux blocs. On déduit alors le déplacement $v(x_0, t)$ correspondant du bloc initial.

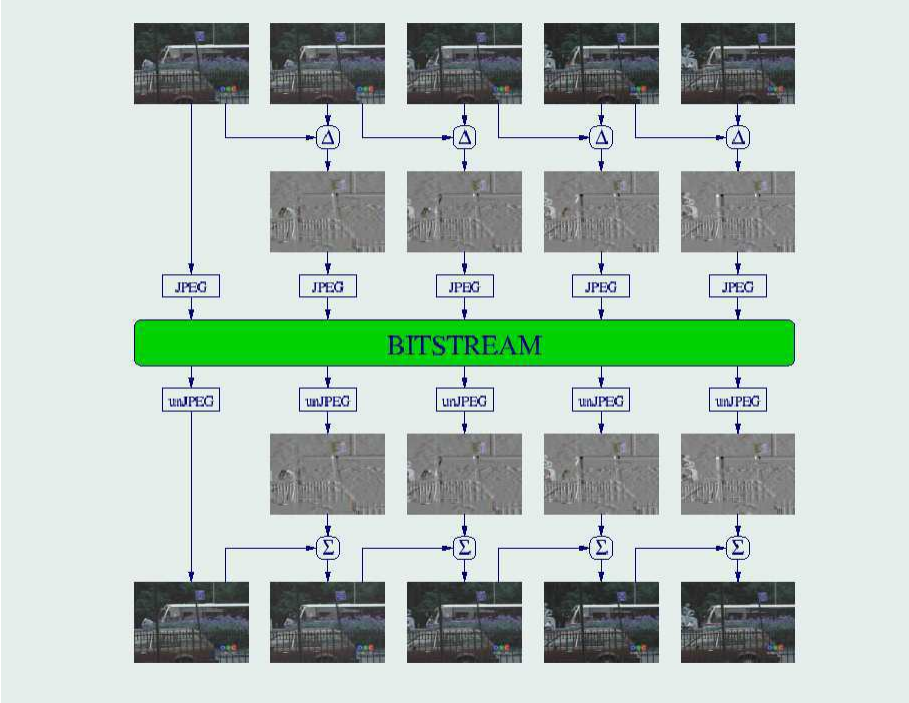


FIG. *1.3 – Codeur/décodeur JPEG-DPCM (Differential Pulse Coded Modulation), ou modulation différentielle des images par impulsions codées (MIC).*

Codeur hybride MPEG1/MPEG2

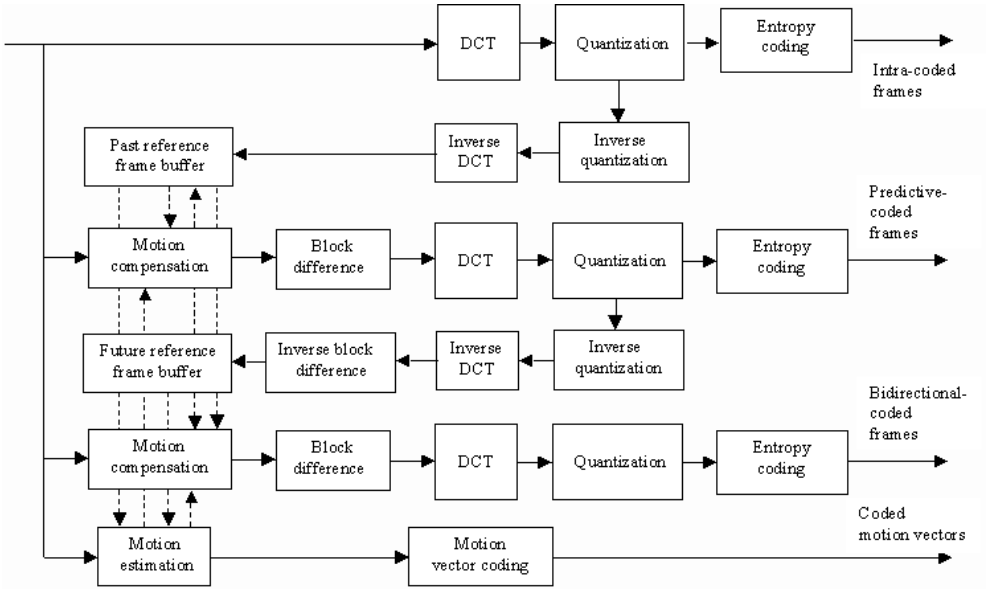


FIG. *1.4 – Schéma de codage MPEG1 et MPEG2. DCT, quantification, estimation de mouvement de type BM puis émission des quatre flux d'information comprenant les trames I, P et B, et les vecteurs de mouvement*

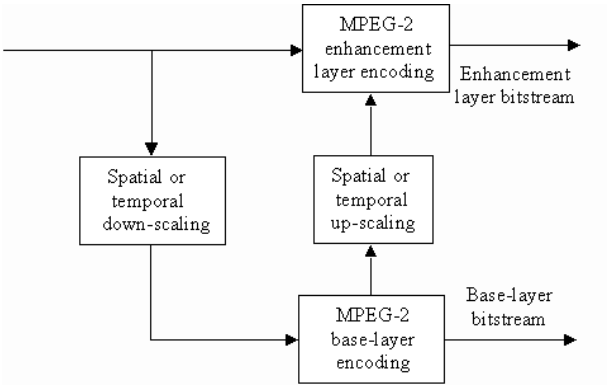


FIG. *1.5 – Codage de l'échelonnabilité sur un codeur MPEG2 (non-existante sur MPEG1)*

Le codeur hybride DVC basé sur la DWT

Le codeur DVC du Heinrich-Hertz Institute (HHI), développé par D. Marpe, Th. Wiegand et H.L. Cycon [MWC02], est basé sur une transformée orthogonale (DWT Video Codeur). Ce codeur utilise une compensation de mouvement par blocs, lesquels présentent une superposition (OBMC, Overlapped Block-Motion Compensation). Ce schéma a été proposé dans le cadre d'un développement du standard MPEG4. Ses performances, annoncées en 2002, sont supérieures à celles du codeur MPEG4 précédent et équivalentes au codeur H26L (versions TML 8/9). Il a été évalué en troisième place à Sidney (2001) derrière le H26L, dans une version ne comportant pas encore le codage des trames B (bidirectionnelles). Ce codeur utilise une nouvelle famille d'ondelettes biorthogonales de Petukhov [MHCP00], à paramètre unique (filtres IIR). Celles-ci présentent davantage de degrés de liberté, dans leur conception, que les filtres biorthogonaux FIR classiques et de meilleures performances que les filtres 9/7 (base d'empreintes du FBI).

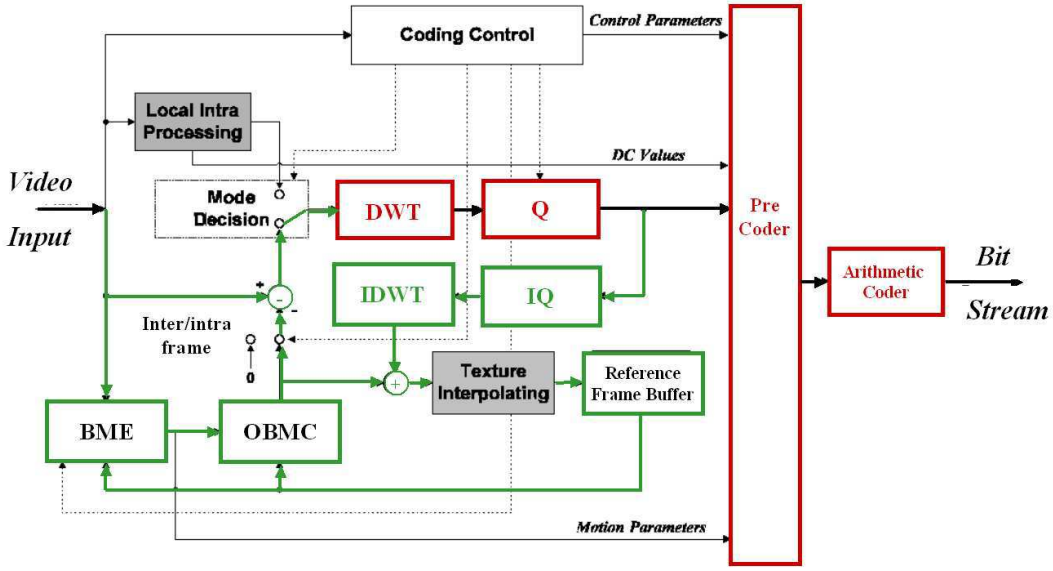


FIG. *1.6 – Le codeur hybride DVC de HHI (Marpe, Wiegand), basé sur une OWT.*

1.2.5 Codeurs 2D+T

La compression est réalisée “en bloc” sur une séquence spatio-temporelle avec ou sans compensation de mouvement. Cette compression peut se faire sur le bloc 3D (ou 2D+T) complet (OWT3D) ou en deux étapes : OWT2D dans le domaine spatial, suivi d’une OWT1D sur l’axe temporel. Dans les deux cas il est possible d’utiliser des ondelettes.

Les codeurs 2D+T à ondelettes

Des codeurs 2D+T ont été développés sur la base d’une transformée en ondelettes. Dans ces codeurs la compensation de mouvement est, ou n’est pas, réalisée. Le codage se fait sur une partie de la séquence, ou GOF (Group Of Frames), qui est alors considérée comme un bloc 3D.
Dans le codage sans compensation de mouvement la décorrélation temporelle, aussi bien que spatiale, peut être réalisée soit par une transformée tri-dimensionnelle et des ondelettes séparables biorthogonales $7-9 \times 7-9$ [Karlsson, Vetterli], soit par une transformée bidimensionnelle, dans le champ spatial, puis suivie d’une transformation selon l’axe temporel.

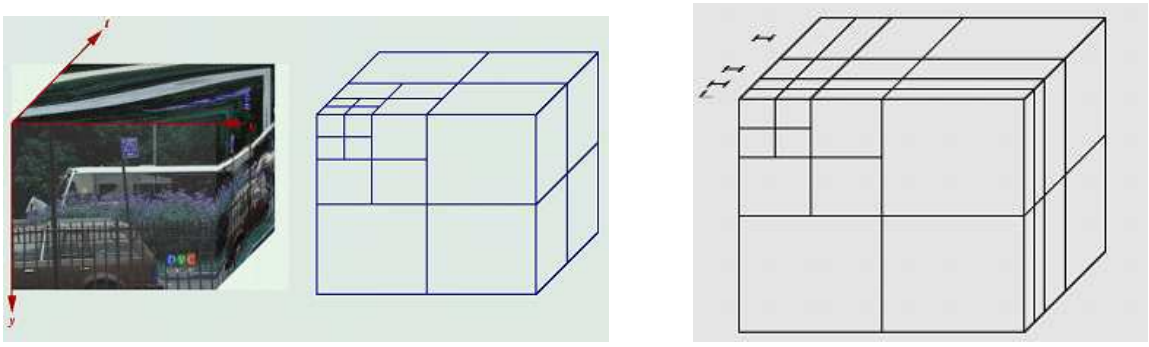


FIG. *1.7 – a) Codeur 2D+T par ondelettes sur un groupe d'image pris comme un bloc 3D. b) Codeur par ondelettes en deux phases : codage 2D spatial, puis codage 1D temporel*

Les codeurs 2D+T à ondelettes et MC

Des codeurs 2D+T à ondelettes possédant une compensation de mouvement ont été réalisés sur la base du schéma lifting de Wim Sweldens [Swe95, DS98].

Principe du schéma lifting : Le lifting est une implémentation de la TO qui permet de décrire la transformée sur des supports de forme arbitraire (création d'ondelettes non-linéaires). :
- Transformée le long de trajectoires d'objets (en temps) : codage 2D+t compensé en mouvement.
- Transformée sur le support des objets : codage par objets.

Le lifting est une série d'opérations qui consiste à transformer un jeu d'échantillons par :
- Séparation des échantillons en éléments pairs (e_n) et impairs (o_n)
- Action d'un jeu d'échantillons sur l'autre, par :
- prédiction : $e = e - f(o)$
- mise à jour (update) : $o = o - g(o)$
- d'autres opérations inversibles de décimation et mise à l'échelle

On part d'un échantillon x_n à la position n, que l'on décompose en un échantillon pair x_{2n} et un impair x_{2n+1}. Une prédiction des coefficients impairs est alors faite à partir des sites pairs, puis une mise à jour (correction) des coefficients pairs est réalisée à partir des vraies valeurs des coefficients impairs.
Le schéma lifting permet de décrire la TO sur des supports de forme arbitraire (création d'ondelettes non-linéaires). On peut ainsi réaliser des TO adaptées à des trajectoires d'objets, ce qui permet la compensation de mouvement 2D+T (C.Guillemot [VG03]. La TO peut aussi être appliquée au "support" des objets (codage de l'objet).

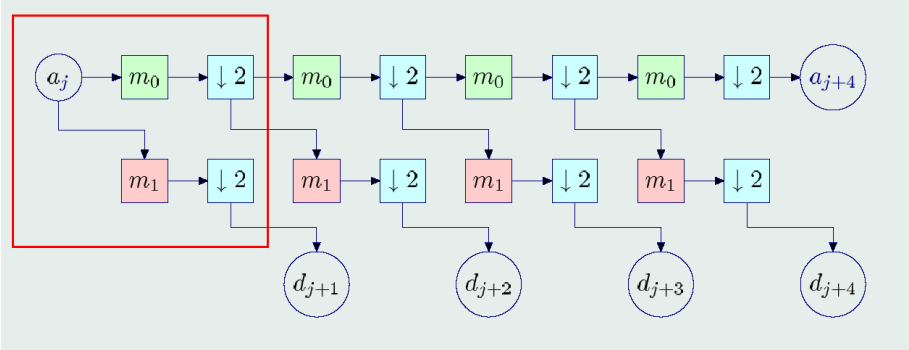


FIG. *1.8 – Etage de la DWT pour le schéma lifting*

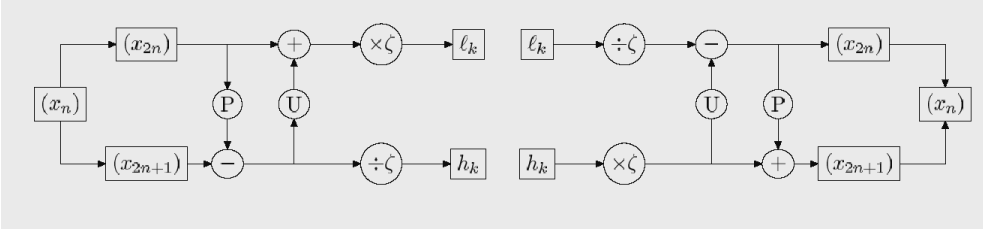
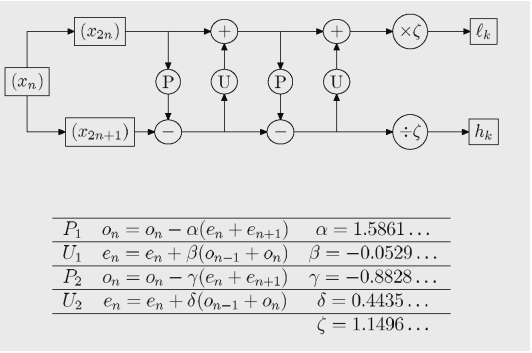


FIG. *1.9 – Lifting : transformées directe et inverse*



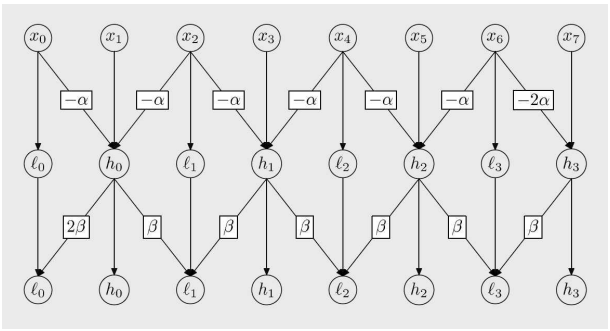


FIG. *1.10 – a) Schéma lifting sur une transformée $7-9$: La prédiction est faite, à partir des coefficients pairs, sur les coefficients impairs ; la mise à jour (ou rebouclage) est faite ensuite sur les coefficients pairs b) Transformée $7-9$ un domaine borné : le schéma de propagation en "papillon" est replié sur les bords ; la méthode reste identique pour la prédiction et la remise à jour (Update)*

Codeur 2D+T à ondelettes et MC : 3D-ESCOT

Le codeur 3D-ESCOT (Embedded subband coding with optimal truncation) de Xu, Li, Zhang et Xiong (Microsoft Research, Texas A&M) [XLZX00], est basé sur la version du codeur statique EBCOT (JPEG2000) de Taubman [TZ94, Tau00] et développé pour la compression de séquences (3D ou "2D+T"). La transformée spatio-temporelle de ce codeur calcule des vecteurs de mouvement entiers et utilise le "motion-threading" c'est-à -dire le parcours de chaque pixel au cours du temps. Cette méthode permet de partitionner un bloc spatio-temporel de pixels en fils (threads) reliant les pixels. La transformée temporelle utilise un schéma lifting avec des ondelettes $7-9$ sur les "motion threads". La transformée spatiale utilise des ondelettes 7-9 sans lifting sur support de forme quelconque (pour un codage par objets).

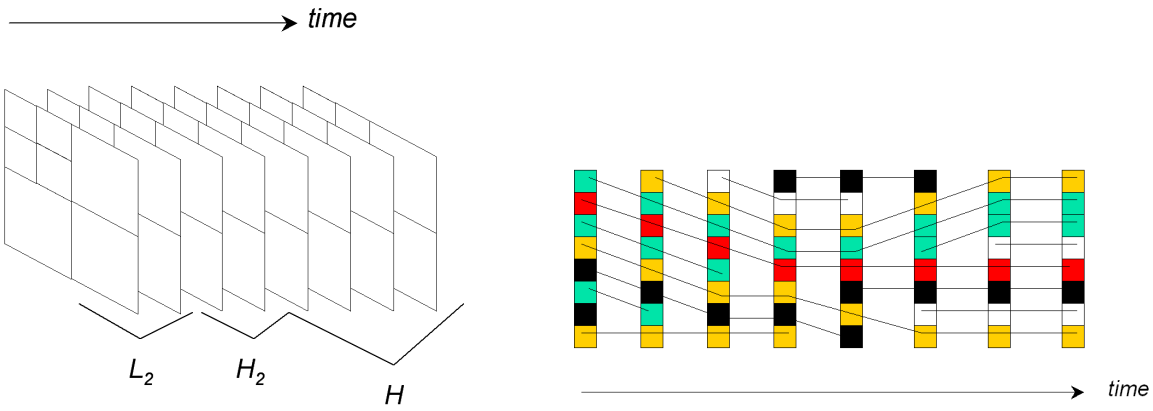


FIG. *1.11 – a) Codage d'un GOF par DWT-2D suivi d'une DWT-1D appliquée sur l'axe temporel b) Codage par motion-threading utilisé dans le schéma du codeur 3D-ESCOT. La DWT-1D est appliquée non plus de façon dyadique, selon l'axe spatial puis temporel, mais selon les liens unissant un même pixel en mouvement (motion threading). Le déplacement de chaque pixel est suivi sur l'axe temporel et sa trajectoire est codée par une DWT-1D avec schéma lifting.*

Codeur 2D+T à ondelettes et MC : Codeur de Woods

Ce codeur fonctionne sur la base d'un schéma lifting [WL01]. Il est issu d'un groupe de travail MPEG4. C'est un codeur échelonnable en temps, en résolution et en SNR. La transformée est compensée par des *mouvements non-entiers*. Pour cela le développement de MCTF (Motion-Compensated Temporal Filtering) a été réalisé au demi-pixel. La transformée spatiale est identique et le codage entropique est réalisé par arbre de zéros (au lieu du codage intra-bande).

1.2.6 Codeur sans transmission des VM (vecteurs mouvement)

Un nouveau type de codeur vidéo développé par C.M. Lee et al. [Lee04], utilisant pour la transmission un code détecteur/correcteur d'erreur de type BCH (Bose-Chaudhuri-Hocquenghem), a permis récemment de coder une séquence vidéo sans transmission des vecteurs de mouvement. La méthode consiste à insérer des "0" dans le spectre de Fourier de l'image, ou plus exactement au niveau des hautes fréquences des spectres des blocs de l'image partitionnée en blocs 13x13. Cette opération revient à effectuer un codage BCH. A la réception il est possible de décoder les vecteurs de mouvement grâce à la redondance introduite dans chaque bloc (dont on a "augmenté" la taille de 13 à 16 à l'émission) . Un test réalisé sur des codeurs H263+ et H264 a montré que les vecteurs de mouvement peuvent être retrouvés au niveau du décodeur sans être transmis explicitement.

1.3 Les normes vidéo

Deux organismes de standardisation coexistent : ISO (MPEG1, MPEG2, MPEG4) et ITU (H262, H263, H263+, H26L). Ces deux organismes se sont regroupés pour former le JVT (Joint Video Team) qui a développé la norme H264.

1.3.1 Différences essentielles

La norme MPEG1 représente un des premiers standards d'un codeur hybride. Celui-ci est basé sur un découpage spatial des trames en blocs 8×8 avec codage de chacun par DCT et sur une décorrélation temporelle par blocs (BM). Les images peuvent être entièrement codées spatialement (trames "I" ou INTRA), mais aussi prédites (trames "P"), par compensation de mouvement, et par "interpolation" (trames "B" ou bidirectionnelles, voir schéma 1.4). Le train binaire est ainsi transmis par GOP et selon une schéma de successions de trames adapté à la nécessité de corriger plus ou moins souvent les images compensées. Le GOP correspond par exemple à une série "IPPB-PIPPB.....". Si les erreurs de compensation sont trop importantes, il est nécessaire de compenser plus souvent et l'on transmet alors plus souvent une trame complète "I". Pour revenir au "bits-tream" transmis, la redondance spatiale est encore réduite par quantification scalaire et codage entropique (Huffman). Le débit binaire de $1,5$Mbits/s atteint, permet la compression sur disque vidéo. La norme MPEG2 est une extension de la norme MPEG1 à des débits variant de $1,5$ à 30Mbits/s pour des applications de télédiffusion. De plus le codage de MPEG2 est échelonnable (voir fig. 1.5).

1.3.2 Codeur MPEG4

Le développement de codeurs "orientés objet" a démarré avec la norme MPEG4. Les grandes lignes qui caractérisent cette norme aux ambitions larges sont les suivantes :
- adaptativité à la résolution, à la qualité, à la complexité (de décodage).
- Environnement virtuel (synthèse)

- Segmentation objet (maillage) et différentiation des codages Objet/Sprite (région animée/statique).
La norme MPEG4 est une norme totalement ouverte et sa normalisation peut prendre au moins une dizaine d'années. L'approche Java (MPEGJ) permet a priori de n'avoir aucun schéma de codage ou compression figé, puisque le code de décodage d'un objet peut se trouver intégré dans une applet Java contenant le codage de l'objet . On peut dès lors envisager tous types de schémas à base d'ondelettes, d'ondelettes spatio-temporelles, d'ondelettes redondantes (CWT). Les méthodes entreront alors en concurrence en fonction de leur efficacité de codage et de décodage. Plusieurs méthodes permettent la reconnaissance et le codage des points de repère du visage ou du corps dans sa totalité. Il en est de même pour les arrière-plans quasi-statiques (sprites).
La structure de codage implique le codage de forme pour les VO (Video Object), segmentés arbitrairement, et la compensation de mouvement ainsi que le codage de texture basé DCT (8x8 DCT ou "shape-adaptive" DCT). Un avantage majeur du codage basé contenu de MPEG4 est que l'efficacité de compression est améliorée de façon significative pour certaines séquences video en utilisant des outils spécifiques de prédiction de mouvement basés objet pour chacun des objets d'une scène.
Plusieurs techniques de prédiction de mouvement peuvent être utilisées pour obtenir un codage efficace ainsi qu'une représentation flexible des objets.

– Estimation et compensation de mouvement standard 8x8 ou 16x16
– Compensation de mouvement globale basée sur la transmission d'un "sprite" statique. Un sprite statique peut se présenter comme une image non-animée de grande taille décrivant de façon panoramique le fond *complet* d'une scène. Ce "sprite panorama" est transmis une seule fois au décodeur lors du premier "frame" de la séquence et ce pour toute la séquence. Ensuite, pour chaque image consécutive de la séquence, seuls 8 paramètres de mouvement globaux décrivant le mouvement de la caméra sont codés afin de reconstruire l'objet. Ces paramètres représentent la transformée affine à appliquer au sprite transmis dans le premier frame pour retrouver le fond de chaque frame de la séquence (fig. 1.12).

1.3.3 AVC H264 et MPEG4-visual part 10

L'émergence de l'AVC "Advanced Video Coding", issu du regroupement de l'ISO et de l'ITU en JVT (voir ci-dessus), et connu sous les noms "ITU/H264" ainsi que "ISO/IEC/MPEG4 part 10", a permis, entre autres, une plus grande souplesse que le codage par blocs classique. L'AVC comporte notamment la possibilité de regroupement des blocs élémentaires en "macroblocs" de taille et de forme variable (bandes, rectangles, carrés, de résolution dimensions variables). En ce sens une certaine souplesse est donnée dans le sens contextuel du codage spatial. De même pour le codage temporel, la possibilité est donnée de rechercher un bloc similaire se trouvant à une distance temporelle beaucoup plus grande (jusqu'à une distance de 16 images). Cette norme est actuellement la plus intéressante pour les développeurs de codeurs vidéo ; elle représente, dans le courant des codeurs hybrides par blocs, un assouplissement et un développement dans la structure de ce type de codeur. Ce retour de la norme MPEG4, porteuse d'une définition très ambitieuse et avant-gardiste de 19 "profiles", mais de mise en oeuvre extrêmement difficile montre une certaine frilosité à l'égard des développements complexes. Il montre aussi un certain réalisme vis-à-vis des difficultés techniques. L'actuelle H264 n'est plus porteuse que de 3 des "profiles" de MPEG4 part 2. Il semble néanmoins

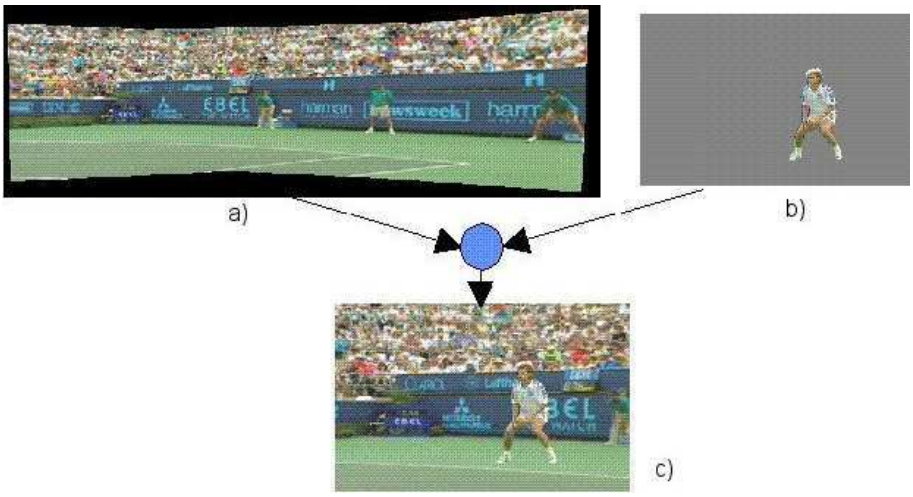


FIG. *1.12 – Exemple de codage de sprite : La Séquence Edberg. a) arrière scène = sprite panorama b) avant scène = objet segmenté c) frame reconstruit à partir du sprite a + les paramètres de caméra et de l'objet b*

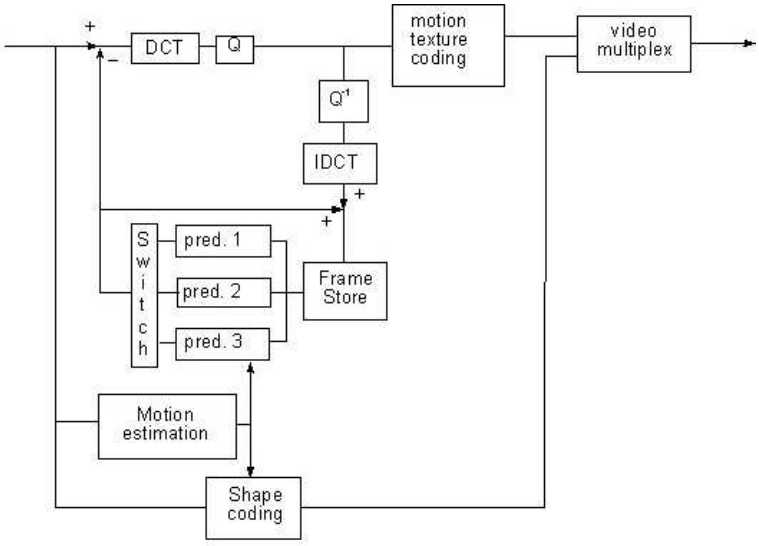


FIG. *1.13 – codeur MPEG4*

dommageable que l'effort vers ce qui parait assez logiquement brosser les traits des futurs codeurs vidéo, à savoir la prise en compte de l'information dans son sens contextuel (compréhension de la scène et adaptativité du codage), n'ait pas entraîné davantage d'intérêt de la part des organismes

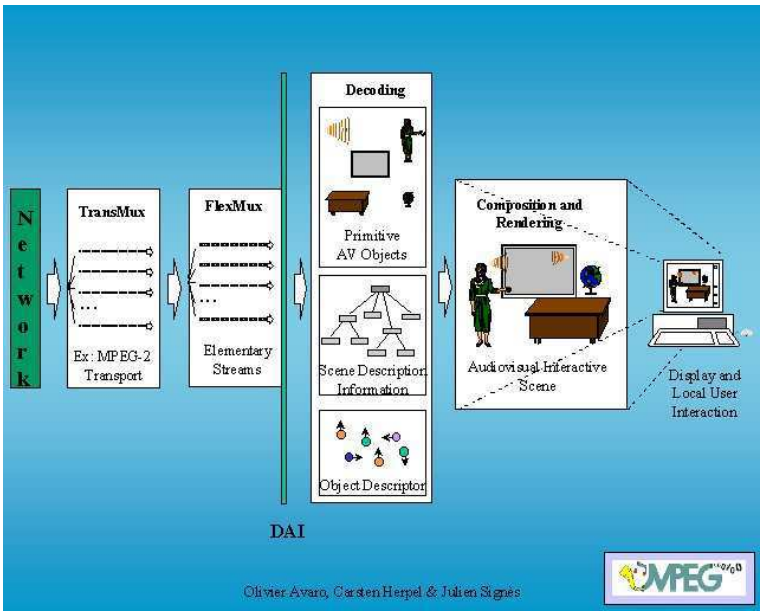


FIG. *1.14 – Forward chanel MPEG*

de recherche amont dans ce domaine.

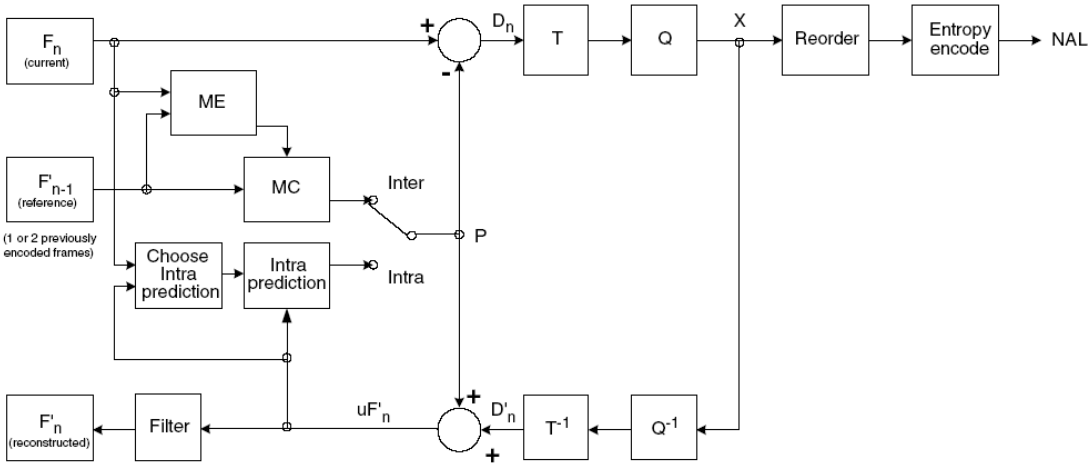


FIG. *1.15 – Schéma du codeur H264*

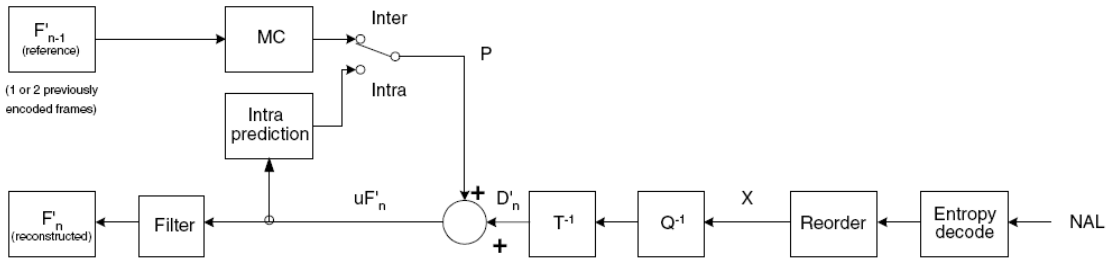


FIG. *1.16 – Schéma du décodeur H264*

Chapitre 2

Estimation de mouvement : Etat de l'art

L'estimation de mouvement (EM) a été étudiée par de très nombreux auteurs parmi lesquels Wang, Weiss, Bergen et Adelson [AB85, WA94a, WA94b], Heeger [Hee87], Horn et Schunk [HS81], Weber et Malik [WM95], Wu et Kanade [WKCL98], et plus récemment Leduc [LMMS00], Bernard [Ber99b], Lee [Lee04].
Précisons tout d'abord que l'estimation de mouvement est une quantification du mouvement simple, par vecteurs de translation, orientés pixel bloc ou objet, ou plus complexe, mettant en oeuvre des méthodes de calcul de trajectoire dans des systèmes bouclés ou non. Les applications de l'E.M. sont surtout la réduction de la redondance temporelle pour la compression et l'analyse de scène. Quatre méthodes reviennent principalement dans l'EM. Ce sont :
- La mise en correspondance de blocs
- Le filtrage spatio-temporel (avec ou sans compensation)
- La mesure de champ dense ou flot (flux) optique
- La non transmission de l'information de mouvement. Cette dernière méthode, dont nous avons parlé dans la présentation d'un codeur sans transmission de vecteurs de mouvement (Lee C.M. [Lee04]) sous-entend bien sûr que, dans une chaîne de transmission vidéo par exemple, le codage du mouvement est fait sous une autre forme que la transmission "explicite" du mouvement. L'information transmise permet de retrouver le champ de mouvement au niveau du décodeur. Seule différence avec les trois premières méthodes : on ne cherche pas à quantifier l'information préalablement au traitement. L'utilisation est bien sûr ici la compression uniquement et non l'analyse (immédiate) de scène.

2.1 La mise en correspondance de blocs (B.M, block matching)

Dans la méthode d'appariement par correspondance de blocs, on recherche la "meilleure correspondance" entre deux fenêtres spatiales (ou blocs) situées dans des trames consécutives . Cette correspondance peut se faire par minimisation de l'erreur quadratique entre les illuminations correspondant aux pixels respectifs des deux blocs. On déduit alors le déplacement v(x0 ; t) correspon-

dant du bloc initial. La mise en correspondance de blocs ou Block-Matching (BM) est comme nous l'avons vu l'une des méthodes de réduction de la redondance temporelle. Elle consiste à découper l'image en blocs, ou carrés, de 8x8 ou 16x16 pixels, puis à rechercher comment se déplacent ces blocs entre deux trames successives n et n+1 (Fig. 2.1). On recherche dans la trame n, précédant celle dans laquelle on se trouve (n+1), le *bloc le plus similaire*. On considère alors qu'il est inutile de refaire le codage spatial puis l'émission du bloc n : il suffit d'envoyer le "vecteur de mouvement" correspondant au déplacement du bloc du frame n au frame n+1. La similitude entre les blocs est donnée par la différence entre les *luminances moyennes* calculées sur chacun des blocs. La méthode est un peu raffinée dans la mesure où l'on ne recherche pas un bloc similaire dans n'importe quel partie de l'image mais dans un voisinage (ordres (1 + 2) ou 4) du bloc de la trame n passée. le parcours du voisinage peut se faire aussi dans un ordre particulier (en spirale dextrogyre par exemple, comme pour H26L).

Cette méthode de block matching est la méthode la plus basique mais aussi une des plus efficaces et anciennes pour estimer le mouvement. En revanche elle ne fait appel à aucune information contextuelle dans l'image et n'a aucune signification dans ce sens. Elle représente la forme la plus élémentaire d'estimation de mouvement par un *champ de vecteurs de translation* entre deux trames. Un assouplissement et une amélioration a été apportée récemment dans les codeurs hybrides comme H26L et H264 en ouvrant le voisinage de recherche du bloc à une plus grande étendue spatiale et à une dépendance temporelle plus grande. Ainsi il est possible de rechercher dans un passé temporel de 16 trames un bloc équivalent. Dans la séquence "tempête" (téléchargeable sur [**?**]) le codeur peut rechercher un objet déjà codé jusqu'à 16 trames en arrière, par exemple une des feuilles mortes.

2.2 Le filtrage spatio-temporel

Le filtrage spatio-temporel présente en premier lieu l'intérêt de ne pas nécessiter de rebouclage. Ceci évite l'accumulation d'une erreur d'estimation d'une trame à l'autre, jusqu'à une augmentation importante de cette erreur et à un "décrochage" complet de la compensation qui nécessite alors un recodage complet d'une trame Intra. Dans les applications de poursuite, ou d'analyse de scène, ce même filtrage peut, en revanche, être inséré dans un rebouclage de type Kalman [MLMS97]. Ce filtrage peut donc être réalisé "au fil de l'eau", de préférence sur un champ dense (flot optique). Ce filtrage se fait donc sur la base du mouvement au niveau pixellique (et éventuellement d'objets) et permet d'obtenir une estimation relativement précise et non fluctuante.

2.3 Le flot optique

2.3.1 Présentation

Le problème posé est celui de la mesure du mouvement dans une séquence d'images vidéo. En général, l'évolution de l'image au cours du temps est due principalement a deux facteurs :
- des sauts entre deux séquences successives, qui sont rares et ponctuels
- le déplacement relatif des objets filmés et de la caméra.

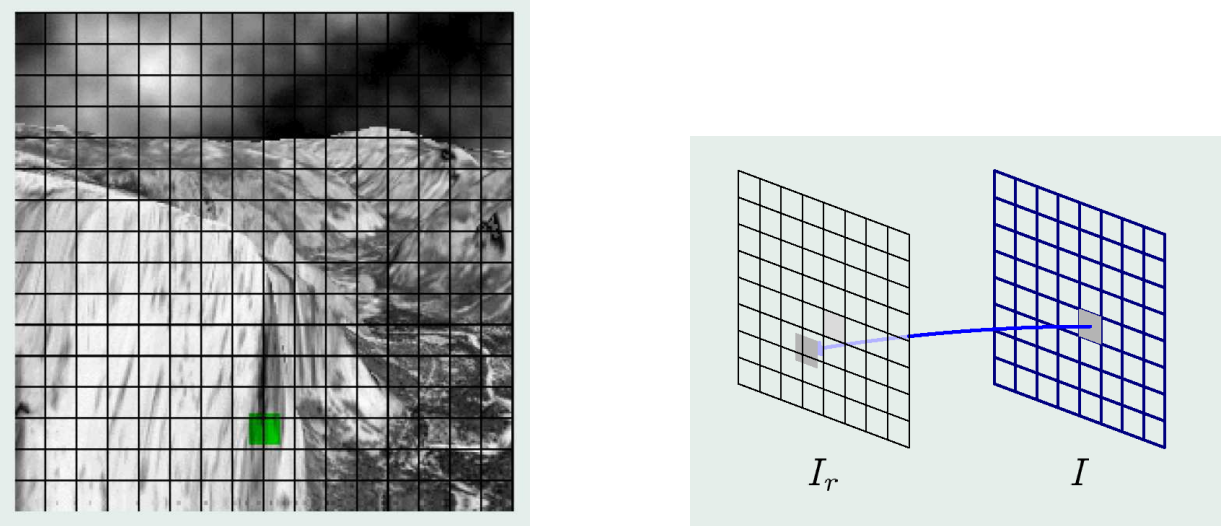


FIG. *2.1 – Block matching de base sur un voisinage de la taille du bloc. Pour chaque bloc de pixels de taille $a \times a$ initialement à la position (x_i, y_i) on recherche un bloc similaire dans une fenêtre W_i de la trame suivante. La fenêtre correspond en général à un voisinage proche (MPEG2) : $W_i = (x_i \pm a, y_i \pm a)$ a) Découpage de l'image en blocs et positionnement d'un bloc similaire dans la nouvelle trame. b) Recherche d'un bloc similaire entre l'image de référence et la nouvelle trame.*

Le mouvement relatif des objets et de la caméra est un champ de vecteurs, à trois composantes, des vitesses des objets filmés dans le référentiel de la caméra. Ce champ de vecteur correspond au mouvement réel. La scène étant projetée sur le plan du film de la caméra, on définit sur le plan-film (ou plan focal image) de la caméra un deuxième champ de vitesse qui est le *champ de vitesses projeté*. On note p l'opérateur de projection (qui peut être linéaire ou non). Pour chaque point x de l'image, qui est le projeté $p(X)$ d'un point réel X de vitesse V , le flot optique en x est alors le vecteur $v = dp(X)V$. L'objet de la mesure du flot optique est d'estimer le flot optique sur la base d'une séquence d'images filmées I(t ; x).

La mesure du flot optique a un certain nombre d'applications possibles. Elle peut servir en tant que telle pour faire de la compression de séquences d'images vidéo par compensation de mouvement (prédiction d'images sur la base d'un champ de déplacement). La mesure du flot optique sert également à l'analyse de scènes : le mouvement apparent des objets d'une scène peut permettre de reconstruire une scène tridimensionnelle si on dispose d'informations supplémentaires sur la nature du mouvement réel. Ces techniques servent donc pour la construction de modèles tridimensionnels d'objets réels (acquisition tridimensionnelle) pour la réalité virtuelle, ou encore en robotique, pour construire une représentation de l'environnement d'un robot qui se déplace.

2.3.2 La mesure du mouvement

Le point réel d'un objet mobile dans l'espace est défini par $X(t) = \{X_1(t), X_2(t), X_3(t)\}$. Si l'on observe cette scène dans une séquence d'images animée, la projection $x(t) = (x_1(t), x_2(t))$ du point $X(t)$ se déplace sur le plan image. Le déplacement de l'ensemble des points image est défini par le champ de déplacement visuel appelé encore *flot optique*. Chaque point de l'image est déterminé, au temps précis t, par sa "fonction couleur" (luminance + chrominance) $I_t(x(t))$. La fonction $I(t, x(t))$ du point image $x(t)$ permet donc de caractériser entièrement celui-ci par sa couleur et sa vitesse. La valeur du flot optique au temps t et pour un point image $x(t)$, est définie comme la vitesse de déplacement du point image :

$$v = (v_1, v_2) = (\frac{dx_1}{dt}, \frac{dx_2}{dt}) \tag{2.1}$$

La détermination du flot optique est rendue difficile par les variations, locales ou totale, de l'éclairement d'une scène au cours du temps. On fait donc dans un premier temps le choix d'une hypothèse simplificatrice qui consiste à considérer l'illumination constante sur l'image au cours de la séquence analysée.
Cette hypothèse d'illumination constante se traduit par une indépendance de $I(t, x(t))$ par rapport au temps, c'est-à-dire par :

$$I(t, x(t)) = I_0 = cte \tag{2.2}$$

La *dérivation particulaire* de la fonction couleur $I(t, x(t))$ donne donc, d'après [2.2] :

$$\frac{d}{dt} I(t; x(t)) = \left[\frac{\partial I}{\partial t} + \frac{\partial I}{\partial x}.\frac{dx}{dt}\right] = \left[\frac{\partial I}{\partial t} + \frac{\partial I}{\partial x_1}.\frac{dx_1}{dt} + \frac{\partial I}{\partial x_2}.\frac{dx_2}{dt}\right] = 0 \tag{2.3}$$

Que l'on peut encore écrire :

$$\frac{\partial I}{\partial t} + v \cdot \nabla I = 0 \tag{2.4}$$

Cette dérivée particulaire de la fonction couleur $I_t(x_1, x_2)$ le long du flot optique (v_1, v_2) s'écrit encore :

$$\frac{\partial I}{\partial t} + \frac{\partial I}{\partial x_1} v_1 + \frac{\partial I}{\partial x_2} v_2 = 0 \tag{2.5}$$

Cette relation est appelée **équation du flot optique**.
On ne dispose donc que d'une unique contrainte scalaire pour résoudre l'équation [2.2] à deux inconnues, v_1 et v_2 : c'est le **problème d'ouverture** (Fig. 2.2). Un moyen de trouver une solution unique qui résout cette équation est d'introduire une hypothèse supplémentaire. Cette méthode est commune à toutes les résolutions du flot optique. A ce stade, plusieurs méthodes ont été exploitées :
1) Recherche de la solution la plus régulière, ou "régularisation" de Horn et Schunk [HS81]
2) Utilisation de filtres spatio-temporels (filtres de vitesse ou d'accélération) sur l'image d'origine, qui reposent aussi sur le fait que le déplacement est constant sur le support des filtres, autrement

dit une avec une hypothèse de constance locale du flot pour résoudre le système de départ. Cette solution est l'objet des travaux de Fleet, Heeger [FJ90, Hee87], ainsi que Weber et Malik [WM95].
3) spectre spatio-temporel local [AB85]
4) Méthodes d'appariement de blocs (BM)

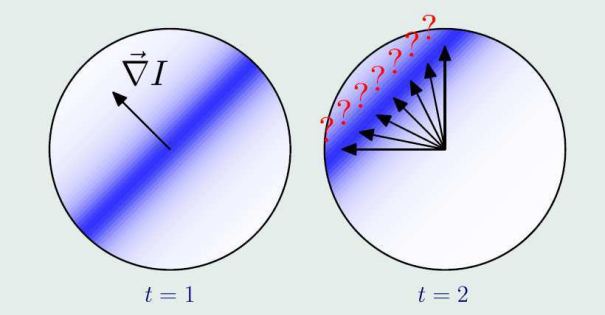


FIG. *2.2 – Le problème d'ouverture : pour une position donnée x l'équation du flot n'admet pas de solution unique (non-unicité) car elle a deux inconnues, les composantes de v. La direction du vecteur vitesse est donc a priori inconnue.*

2.3.3 Un problème mal posé

La mesure du flot optique est un *problème inverse* mais c'est aussi un problème mal posé.
Un problème inverse est dit "mal posé" lorsque dans l'équation suivante 2.6, l'opérateur A que l'on souhaiterait inverser est soit :
- non-inversible
- non unitaire
- non stable (exemple : mauvais conditionnement de matrice d'inversion qui cause des variations importantes de la solution pour des variations faibles de l'observable à inverser)
Ces trois conditions sont dites *conditions de Hadamard.*

$$Ax = y \qquad \Leftrightarrow \qquad x = A^{-1}y \qquad \Leftrightarrow \qquad y - Ax = 0 \tag{2.6}$$

Une méthode de résolution triviale est de rechercher par exemple une solution x permettant de minimiser la norme L ou L^2 ou L^p du terme $y - Ax$ plutôt que d'en rechercher une solution analytique. Si l'on cherche à minimiser la fonctionnelle "d'adéquation quadratique" (méthode des moindres carrés ou LMS) , la solution s'écrit :

$$x = \underset{\acute{x}}{arg\,min}\,|Ax - y|^2 \tag{2.7}$$

a) Régularisation Ceci revient encore à un problème mal posé car cette fonctionnelle n'est pas définie . Dans un deuxième temps on suppose alors que la solution se trouve dans un espace de Hilbert et on rajoute, au terme d'adéquation, un terme de pénalisation afin de "régulariser" ce dernier.

$$x = \underset{\acute{x}}{arg\,min}\,|Ax - y|^2 + \lambda\,|\acute{x}|_H^2 \tag{2.8}$$

La régularisation comme méthode de calcul du flot optique a été proposée en 1981 par Horn et Schunk [HS81]. Elle consiste à faire l'hypothèse que la "meilleure" solution du système est la solution la plus régulière. On cherche alors à minimiser la fonctionnelle (dite d'"attache aux données") :

$$M[v] = \iint \left(v.\nabla I + \frac{\partial I}{\partial t}\right)^2 dx_1 dx_2 \tag{2.9}$$

ce qui revient à trouver une carte de déplacement v(x) qui annule l'équation de base du flot optique (éclairement constant).
Les auteurs ajoutent, à cette fonctionnelle quadratique, une fonctionnelle dite de "régularisation" (ex. norme de Sobolev) :

$$S[v] = \iint\ |\Delta v|^2 dx_1 dx_2 \tag{2.10}$$

Minimiser la fonctionnelle totale consiste donc à rechercher v tel que :

$$v = arg\,min\left(\iint \left(\frac{\partial I}{\partial t} + v \cdot \nabla I\right)^2 dx_1 dx_2 + \lambda \iint |\Delta v|^2 dx_1 dx_2\right) \tag{2.11}$$

où λ est un paramètre qui permet de fixer la régularité et l'adéquation à la contrainte du flot optique. En pratique la résolution du système régularisé consiste en l'inversion d'un système matriciel de grande dimension $(2 \times N \times M)$.

b) Méthode différentielle filtrée La méthode différentielle filtrée, développée par Weber et Malik [WM95], consiste à appliquer l'équation du flot optique non pas à l'image de départ I mais à l'image filtrée par plusieurs filtres spatio-temporels $f_n(x,t)$.

c) Spectre spatio-temporel Adelson et Bergen [AB85] ont proposé une méthode d'analyse fréquentielle locale du spectre spatio-temporel, sur des portions d'espace-temps de la séquence d'images. Leur travail a consisté à montrer que la perception visuelle du mouvement peut être captée par des filtres passe-bande appropriés. Les filtres construits par Adelson et Bergen permettent de "voir" un motif en déplacement à vitesse constante si le rapport entre leurs fréquences temporelles et spatiales moyennes correspond à la valeur de la vitesse et si la direction de la vitesse et la fréquence spatiale sont suffisamment proches.

2.4 Résolution rapide du flot optique par ondelettes

La solution proposée par C. Bernard et S. Mallat est de projeter l'équation du flot sur une base d'ondelettes en faisant l'hypothèse que le flot est constant. Cette méthode est appelée **estimation différentielle projetée** [Ber99b]. On considère la *famille des ondelettes mères* $\psi^n(x)_{n=1...N}$. On obtient ainsi pour chaque paramètre d'échelle j et pour chaque couple de paramètres $k = (k_1, k_2)$ de position la famille d'ondelettes mères :

$$\psi^n_{jk}(x) = 2^j \psi^n(2^j x - k) \tag{2.12}$$

Le développement de l'équation 2.2 selon la base d'ondelettes $\psi^n_{j,k}(x)$ s'écrit sous la forme du produit scalaire :

$$\iint (\frac{\partial I}{\partial x_1} v_1 + \frac{\partial I}{\partial x_2} v_2 + \frac{\partial I}{\partial t}).\psi^n_{jk}(x)\, dx_1 dx_2 = 0 \tag{2.13}$$

Lorsque l'on fait l'hypothèse supplémentaire de flot constant : $v_1, v_2 = cte$, sur le support des ondelettes, on peut sortir du radical les termes v_1 et v_2 On obtient alors :

$$< \frac{\partial I_t}{\partial x_1}, \psi^n_{jk} > v_1 + < \frac{\partial I_t}{\partial x_2}, \psi^n_{jk} > v_2 + < \frac{\partial I_t}{\partial t}, \psi^n_{jk} >= 0 \tag{2.14}$$

Le troisième terme en dérivée du flot par rapport au temps peut se reécrire, puisque l'ondelette ψ^n_{jk} est indépendante du temps :

$$< \frac{\partial I_t}{\partial x_1}, \psi^n_{jk} > v_1 + < \frac{\partial I_t}{\partial x_2}, \psi^n_{jk} > v_2 + \frac{\partial}{\partial t} < I_t, \psi^n_{jk} >= 0 \tag{2.15}$$

On effectue alors une intégration par parties, en considérant que les produits $I_t.\psi^n_{jk}$ sont nuls, puisque l'ondelette est de moyenne nulle sur son support et que I_t est constant sur ce support. On obtient ainsi un *système de N équations du flot optique* dans lequel les coefficients A, B et C peuvent être calculés par une transformée en ondelettes rapide.

$$\underbrace{< \frac{\partial I_t}{\partial x_1}, \psi^n_{jk} >}_{A} v_1 + \underbrace{< \frac{\partial I_t}{\partial x_2}, \psi^n_{jk} >}_{B} v_2 = \frac{\partial}{\partial t} \underbrace{< I_t, \psi^n_{jk} >}_{C} \tag{2.16}$$

D'autre part les ondelettes mères sont les *produits tensoriels* classiques, utilisés en analyse multirésolution, des fonctions ϕ et ψ, ce qui donne les trois ondelettes mères classiques :

$$\begin{cases} \psi_1(x) &= \psi(x_1).\phi(x_2) \\ \psi_2(x) &= \phi(x_1).\psi(x_2) \\ \psi_3(x) &= \psi(x_1).\psi(x_2) \end{cases} \tag{2.17}$$

Remarque : Les ondelettes *presque analytiques* construites précédemment constituent des "frames", c'est-à-dire des familles de vecteurs $\{\phi_n\}_{n\in\Gamma}$ qui permettent de caractériser tout signal f par ses produits scalaires $\{< f, \phi_n >\}_{n\in\Gamma}$, où Γ est un ensemble d'indices fini ou non, et qui possèdent la propriété suivante (par généralisation de Parseval) :

$$M_1 |f|_2^2 \leqslant \sum_{n=1}^{N} \sum_{j \in Z} \sum_{k \in Z^2} \left| < f, \psi_{jk}^n > \right|^2 \leqslant M_2 |f|_2^2 \tag{2.18}$$

Chapitre 3

Analyse spectrale du mouvement

Ce chapitre est une introduction à l'approche de l'estimation de mouvement par ondelettes spatio-temporelles adaptées au mouvement.

3.1 Analyse du mouvement dans l'espace de Fourier

On présente ici les résultats importants de la transformation de Fourier appliquée à des signaux spatio-temporels. Les mouvements considérés sont : translation et rotation à vitesse et accélération constantes, ainsi que changement d'échelle. On s'intéresse ici à la représentation dans l'espace de Fourier du mouvement d'un objet, afin d'expliciter la différence de distribution de l'énergie entre un objet statique et un objet en mouvement. Dans la section suivante, on considérera cet objet en mouvement analysé par une transformation en ondelette continue et l'on verra comment adapter l'ondelette au même mouvement.
La relation entre la distribution d'énergie d'un objet statique $s(\vec{x},t) = s(\vec{x})$, et sa version en déplacement linéaire $s(\vec{x}-\vec{v}t,t)$ peut se présenter mathématiquement par des paires de Fourier. Pour un objet statique :

$$s(\vec{x},t) = s(\vec{x}) \leftrightarrow \hat{s}(\vec{k})\delta(\omega) \tag{3.1}$$

et pour un objet en déplacement linéaire, donc en effectuant le changement de variable $\xi = (\vec{x}-\vec{v}t)$:

$$s(\vec{x}-\vec{v}t,t) = s(\vec{x},0) \leftrightarrow \hat{s}(\vec{k},\omega+\vec{k}.\vec{v}) \tag{3.2}$$

où le terme $s(\vec{x},0)$ désigne le signal à l'instant initial, donc ayant subi la transformation inverse du mouvement au point t (unwarped signal). Le terme $\vec{v}t$ représente lui une translation constante dans l'espace des positions $\vec{x}$.
Les équations 3.1 et 3.2 font état de la redistribution de l'énergie entre un objet sans mouvement et le même objet en déplacement linéaire. Dans le cas de l'équation 3.1, l'énergie est concentrée dans le plan des fréquences nulles, $\omega = 0$. En présence d'un mouvement linéaire, l'énergie est redistribuée sur un plan défini par :

$$\vec{k}.\vec{v} + \omega = 0 \tag{3.3}$$

soit encore :

$$\left[\vec{k}'\omega\right]\begin{bmatrix}\vec{v}\\1\end{bmatrix} = 0 \tag{3.4}$$

où le symbole prime est l'opérateur de transposition.
La relation 3.4 définit un plan de vélocité perpendiculaire au vecteur $[\vec{v}'\ 1]'$
Lorsque l'objet subit une accélération, l'énergie est étalée autour du plan de vélocité dominant. Une **approximation linéaire** à temps fini court est réalisée par l'utilisation d'une fenêtre $w(t)$. Ce fenêtrage temporel peut être inclus dans le filtre de vélocité. L'accélération en vélocité/s est alors reliée à l'accélération en nombre de vélocité/frame fois le nombre de frame/s.

3.2 Analyse spectrale de rectangles en translation uniforme

Nous commençons par analyser une séquences synthétique composée de rectangles en translation à différentes vitesses, qui sont uniformes. La figure 3.1 montre cinq rectangles dont trois sont en translation horizontale à des vitesses constantes de 1, 3 et 10 pix/fr et deux sont en translation verticale à des vitesses de 1 et 3 pix/fr.

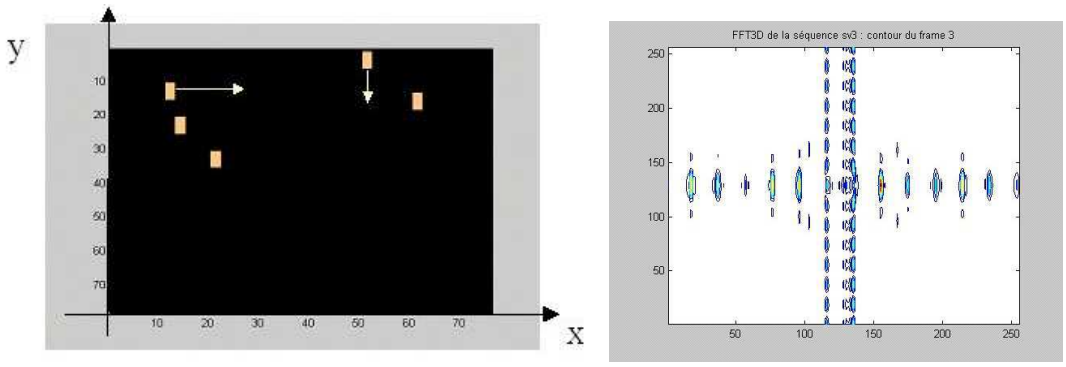


FIG. *3.1 – a) Séquence (*16 *trames) de rectangles en translation horizontale et verticale à différentes vitesses :* $V_h = 1$, *3 et 10 pix/fr et* $V_v = 1$ *et 3 pix/fr. b) Transformée de Fourier 2D d'un rectangle (statique) en représentation "contour".*

On calcule la FFT3d de la séquence puis on affiche les vecteurs d'onde k_y puis k_x en fonction de t respectivement pour les vecteur d'onde $k_x = 3$ et $k_y = 3$ afin d'éviter le plan de fréquence $k = 0$ qui ne comporte aucune information ; Leduc (97) en revanche utilise un filtre non séparable qui ne s'annule pas dans le plan $\omega = 0$.

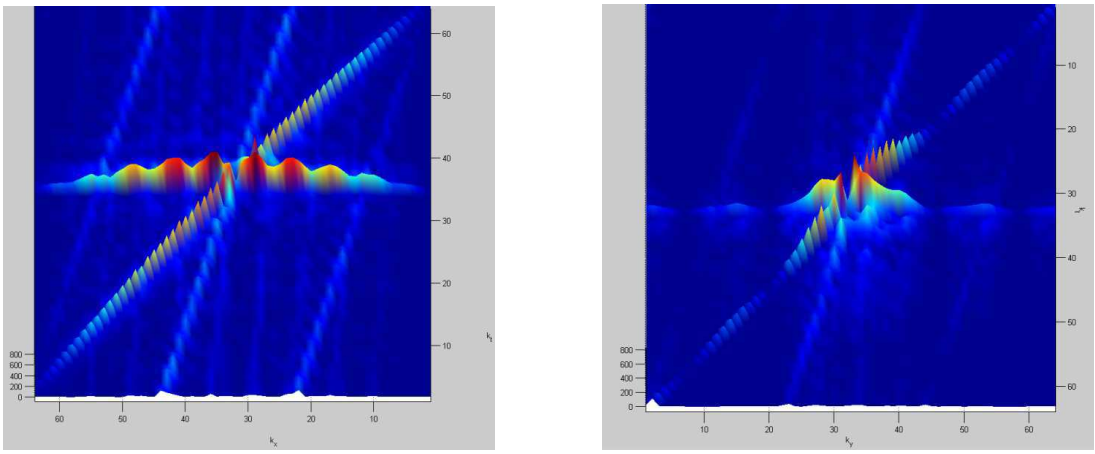


FIG. *3.2 – FFT3D de la scène rectangles en translation uniforme avec $V_h = 1$, 3 et 10 pix/fr et $V_v = 1$ et 3 pix/fr. La TF est calculée sur un GOF de huit trames et le plan spectral central de la FFT3D est affiché a) Affichage selon les vecteurs d'onde (k_x, k_t) qui montre trois familles de droites à trois pentes permettant la détection des trois vitesses horizontales $V_h = 1$, 3 et 10 pix/fr. b) Affichage selon (k_y, k_t) qui montre la même détection pour les deux vitesses verticales $V_v = 1$ et 3 pix/fr*

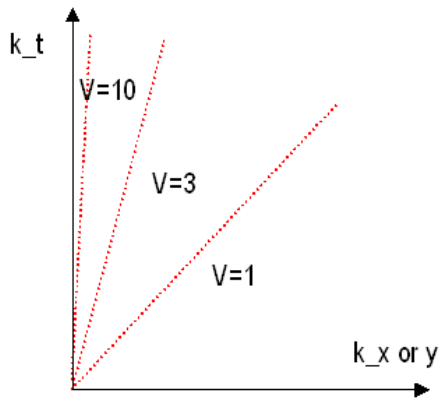


FIG. *3.3 – Repérage des pentes des spectres des trois objets en translation horizontale : les pentes correspondent aux vitesses de 1, 3 et 10 pixels/frame. On remarquera, sur cette figure, que, inversement à l'espace direct (voir Fig. 3.4), plus la composante de la vitesse est élevée, plus cette composante est proche de l'axe du vecteur d'onde temporel et non du vecteur d'onde spatial.*

La figure 3.4 montre cette fois le volume occupé par un rectangle en déplacement uniforme, à une vitesse $c1$, dans l'espace direct. On peut noter que, contrairement au domaine spectral, la droite

correspondant à une vitesse plus élevée ($c = 2$) tend vers l'axe spatial x et non vers l'axe des vecteurs d'onde temporels (Figs. 3.2 et 3.3).

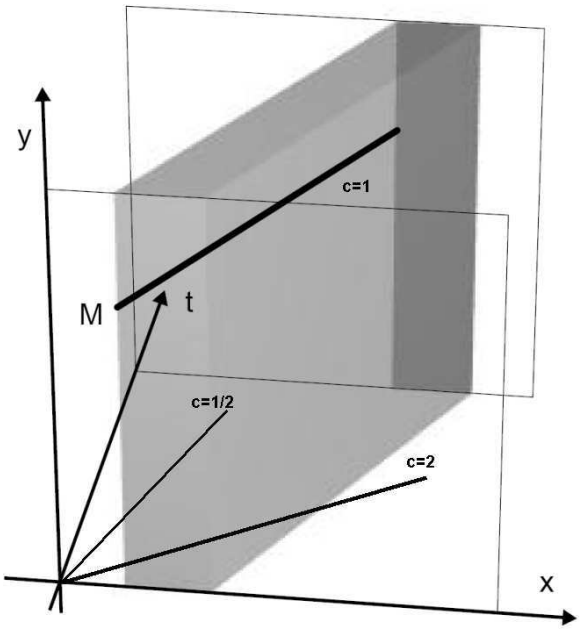


FIG. *3.4 – Volume créé par le déplacement rectiligne selon Ox d'un rectangle dans l'espace 3D spatio-temporel direct à une vitesse* $c_x = 1$. *La représentation montre les trois inclinaisons des droites* $x(t)$ *pour les trois vitesses* $c = 1/2, 1, 2$. *La vitesse* $c = 1/2$ *correspond à la droite* $t = cx$ *de plus forte pente, puisque le rectangle effectue un parcours plus faible dans le même temps par rapport à* $c = 1$.

3.3 Sinusoïdes en translations uniforme et accélérée

La séquence spatio-temporelle 1D+t suivante représente quatre sinusoïdes dont trois sont en translation uniforme à des vitesses différentes et la quatrième est en accélération constante (Fig. 3.5). Les images de la séquence $S_m v(x,t)$ sont de taille 256×256. Le signal $S_m v$ est donné par la relation :

$$S_m v = sin(3x + kt) + sin(5x + 2kt) + sin(8x + .5kt) + sin(16x + .5mt) \tag{3.5}$$

avec $k = 0.2$ le paramètre de vitesse et $m = m + .01$ à chaque nouvelle trame crée l'accélération pour le quatrième terme sinusoïdal. Une transformée de Fourier 3D effectuée sur la séquence complète donne le résultat en Fig. 3.5 dans le plan (k_x, k_y). La figure montre les points du plan (k_x, k_y) correspondant aux trois premières sinusoïdes en mouvement uniforme et la droite formée par la variation de vitesse de la sinusoïde en accélération.

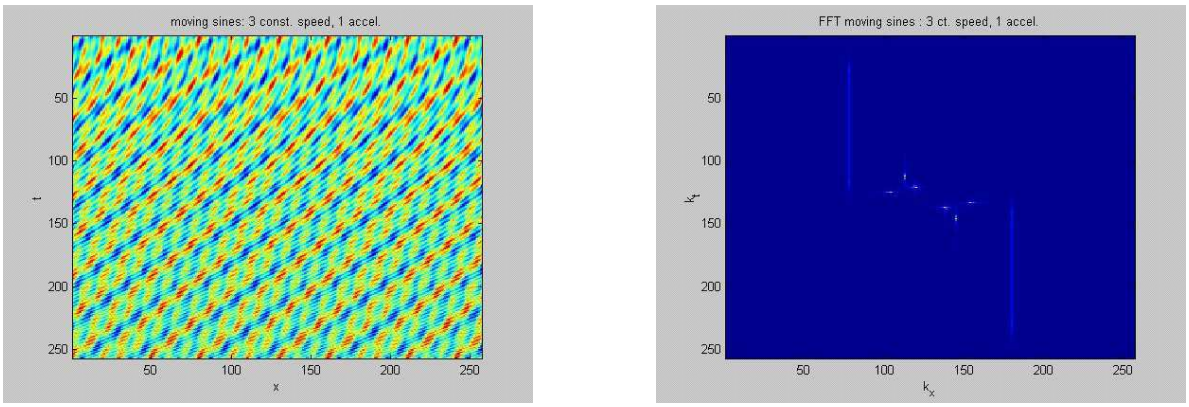


FIG. *3.5 – a) Représentation 3D d'un signal composé de quatre sinusoïdes en translation : trois sont en translation à vitesse constante, la quatrième est en mouvement uniformément accéléré. b) Représentation selon (k_x, k_y) dans le domaine de Fourier des quatre sinusoïdes montrant la détection des 3 sinusoïdes en mouvement uniforme et du trait matérialisant le mouvement accéléré. Une représentation dans les plans (k_x, k_t) et (k_y, k_t) auraient donné des droites avec pente variable comme nous l'avons vu pour le mouvement uniforme. Pour un mouvement accéléré, la droite s'infléchirait, puisque sa pente varie.*

3.4 Décalage dû à la vitesse : représentation dans l'espace 2D+T

3.4.1 Représentation du "plan de vitesses" dans l'espace 2D+T

Nous avons montré sur la figure 3.3 comment se modifie la pente du spectre dans l'espace (kx, kt) en fonction de la vitesse. Lorsque l'on considère un motif se déplaçant dans une direction quelconque dans le plan, sa vitesse $\vec{v}_0 = (v_0x, v_0y)$ place alors le spectre du motif dans un plan défini par les vitesses sur les axes kx et ky. La figure (3.6) suivante représente le plan $\vec{k}\ v_0$ sur lequel est projeté le spectre du motif.

3.4.2 Décalage du spectre dans l'espace 2D+T

La figure 3.7 montre comment se modifie le spectre d'un objet en déplacement, dans un espace 2D+T, à une vitesse $\vec{v}_0$. Le spectre du signal est projeté en $-k\vec{v}_0$, sur le plan que nous venons de décrire dans la figure 3.6, et défini par les droites v_0x et v_0y. La projection de l'axe de l'objet dans ce plan se fait sur la droite Δ de la figure 3.7 (voir aussi [DDM93] pour cette représentation).

3.5 Conclusion à l'analyse spectrale du mouvement

Nous venons de mettre en évidence qu'un mouvement et ses paramètres cinématiques : dérivées d'ordre un, d'ordre deux ou d'ordre supérieur, peuvent être mis en évidence et quantifiés dans une

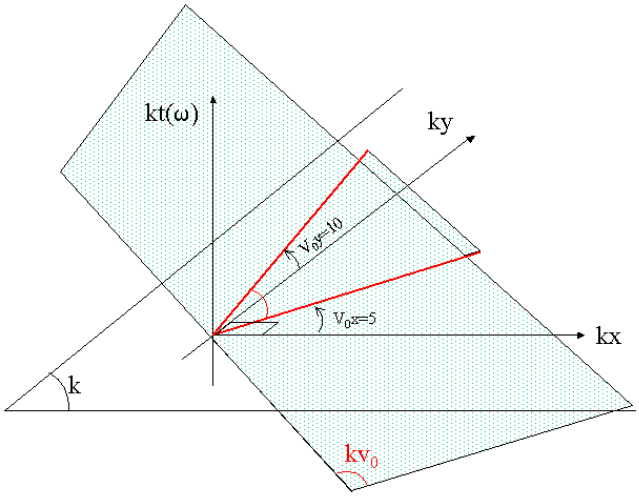


FIG. *3.6 – Plan $\vec{k}$ v_0 dans lequel se décalent les spectres des motifs présentant une vitesse $\vec{v_0} = (v_0x, v_0y)$. L'exemple est pris pour des valeurs approximatives de $v_0x =$ 5 pixels/trame et v_0y le double, soit* 10 *pixels/trame.*

simple analyse de Fourier.
Il est donc possible d'extraire ces paramètres de façon individuelle, dans le domaine spectral, en utilisant un filtrage passe-bande. La détection d'une vitesse, ou, nous le verrons, une "gamme de vitesses", peut donc être réalisée par un filtrage passe-bande. Or les ondelettes sont des filtres passe-bande "scalables" et positionnables dans une image ou une séquence. En d'autres termes, elles peuvent travailler à différentes échelles spatiales et en différentes positions du support de l'objet analysé (donc de l'image) et, nous venons de le voir, de percevoir aussi certaines vitesses, ou plus largement certains paramètres cinématiques d'objets de la scène.

La première remarque est bien évidemment qu'une ondelette "adaptée" à une vitesse ne va "visualiser" correctement que les objets possédant la vitesse pour laquelle elle est adaptée, de la même façon qu'elle le fait pour les fréquences spatiales grâce au paramètre de changement d'échelle a. Il est donc a priori nécessaire d'utiliser une "famille d'ondelettes adaptées" pour analyser par exemple un ensemble de vitesses dans une scène.
Si nous savons a priori que les objets se déplacent entre les vitesses de 1 et 20 pixels par frame, nous choisirons d'utiliser une famille d'ondelettes pouvant analyser les vitesses $1, 3, 6, 12, 20$ pixels/frame avec une sélectivité moyenne. Si en revanche les vitesses qui nous intéressent se rangent dans une gamme de 10 à 50 pixels par frame, nous utiliserons des ondelettes adaptées à des vitesses élevées : $10, 20, 30, 40$ et 50 pixels/frame avec une sélectivité légèrement plus élevée.

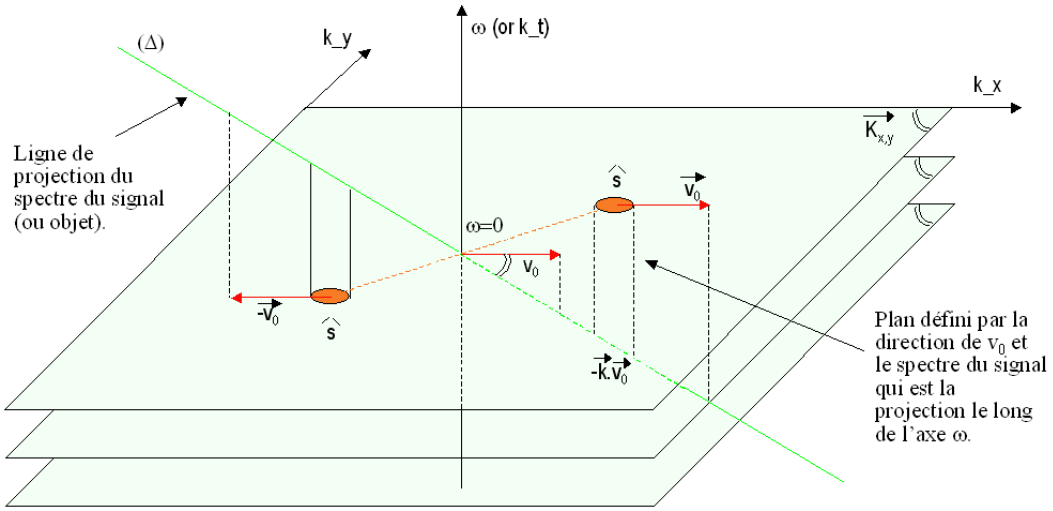


FIG. *3.7 – Décalage du spectre dû à la vitesse d'un objet. La représentation est faite dans le domaine "des vecteurs d'onde spatio-temporels" où chaque plan de vecteurs d'onde (k_x, k_y) correspond à une fréquence temporelle (origine en $\omega = k_t = 0$. Le spectre $\hat{s}$ de l'objet, à l'origine dans le plan $\omega = 0$, est projeté sur la droite Δ, dans les plans $+k\vec{v_0}$ et $-k\vec{v_0}$. Ceci explique comment une ondelette "adaptée" au plan de fréquence $+k\vec{v_0}$ peut permettre de "détecter" des objets possédant une vitesse v_0. Nous verrons plus loin qu'un paramètre de sélectivité ϵ permet de relâcher la contrainte de détection d'une seule vitesse en permettant la détection autour d'un plan de fréquence (ou d'une vitesse) donné.*

Dans le chapitre suivant nous allons montrer comment sont construites ces ondelettes adaptées à la vitesse, quelles sont les conditions d'admissibilité pour une décomposition/ analyse et comment peut varier leur sélectivité. Outre la vitesse nous montrerons aussi quels sont les paramètres de la famille d'ondelettes galiléennes, ondelettes plus particulièrement adaptées à la vitesse, mais aussi à la rotation et à la translation spatio-temporelle.

Chapitre 4

Transformée en ondelettes adaptée au mouvement

4.1 Transformation en ondelettes continue

La décomposition en ondelettes a été développée au départ pour situer de façon plus précise la *position* des fréquences composant le spectre d'un signal. En effet la transformée de Fourier permet de calculer le spectre d'un signal sans donner de précision sur la position des éléments spectraux dans le signal. La transformée de Fourier à fenêtre (STFT ou Short term Fourier transform) offre l'avantage de faire correspondre un spectre à une "fenêtre" du signal. La transformée en ondelettes présente, par rapport à la STFT, l'intérêt d'un aspect multi-échelles de la fenêtre d'analyse grâce à l'utilisation d'atomes temps-fréquences réduits à une "petite onde" au lieu d'une fonction continue comme les bases en cosinus et en sinus utilisées pour la décomposition de Fourier et sa version à fenêtre. Entre la STFT et la TO, on peut citer aussi la transformée de Gabor qui utilise des fenêtres gaussiennes à support limité, mais sans l'aspect multi-échelles des ondelettes, donc sans le changement de fréquence (on dira alors "d'échelle") en plus de la limitation du support.
A l'origine, la TO est construite comme une transformée qui "projette" un signal sur une famille de fonctions qui sont de "petites ondes" car leur support est limité, contrairement à la transformée de Fourier ou de Gabor. Les coefficients qui résultent de cette projection représentent le degré de similitude cette petite onde que l'on translate le long du signal et que l'on dilate. Pour chaque valeur de translation et pour chaque valeur de dilatation (ou "échelle") on crée un nouveau jeu de coefficients. La représentation résultante de l'ensemble des coefficients dans un plan position (ou temps) et échelle s'appelle le "scalogramme". Cette représentation imagée vaut bien sûr pour un signal que l'on a échantillonné. De façon plus mathématique, la transformée en ondelette s'écrit comme un produit de convolution à temps continu et les coefficients sont représentés par les valeurs de l'intégrale ainsi définie dans un espace à une dimension :

$$T_\psi x(a,b) = \int \psi(\frac{x-b}{a})dx =< x, \psi(a,b) > \tag{4.1}$$

La décomposition en ondelettes, au delà de l'ensemble classique à deux paramètres, échelle et trans-

lation, a donné lieu au développement de nombreuses familles à multiples paramètres adaptées, entre autres, au mouvement. Ce sont : les familles de décomposition en ondelettes galiléennes, cinématiques (vélocité), accélérées, rotationnelles, sur la variété ("on manifold"), de déformation, de Schrödinger et relativistes. Ces familles forment des structures de groupes qui présentent des capacité d'analyse beaucoup plus développées que la TO translation-échelle, sont peu connues et présentent a priori l'inconvénient d'être des constructions redondantes. Ceci signifie que les coûts de calcul risquent d'être élevés mais que les performances en analyse sont meilleures que des familles orthogonales où l'information est limitée au minimum, par construction.

4.2 Construction de la T.O. continue "spatio-temporelle"

Le filtrage dans l'espace 2D+T peut se faire dans le domaine direct (appelé couramment "spatio-temporel", ou ST) ou dans le domaine fréquentiel. Nous utiliserons l'appellation "transformation spatio-temporelle directe" pour indiquer à quel moment on parle d'une transformation dans le domaine direct ou "spatio-temporelle spectrale" lorsqu'il s'agira de la même transformation dans le domaine de Fourier.
Nous considérons ici des signaux spatio-temporels à énergie finie. Ce sont :

$$s(x,t) \in L^2(\mathbb{R}^n \times \mathbb{R}, d^n x dt) \tag{4.2}$$

avec

$$|s(x,t)|^2 = \int_{\mathbb{R}^n \times \mathbb{R}} |s(x,t)|^2 d^n x dt < \infty \tag{4.3}$$

4.2.1 Définition de la CWT spatio-temporelle

Une séquence vidéo est un objet à trois dimensions : spatiales (2D) et temporelle (1D). Pour analyser cette séquence, il faut d'abord exprimer la transformée en ondelettes dans le domaine spatio-temporel. Cette transformation (2D+t) est obtenue par produit de convolution (ou produit scalaire dans Fourier) entre la séquence et une famille d'ondelettes à trois dimensions $\psi(x,y,t)$, ou $\hat{\psi}(kx,ky,\omega)$; par la suite on adoptera la notation ω ou k_t lorsqu'on voudra exprimer le vecteur d'onde dans la direction temporelle.
La CWT 2D+t dans le domaine spatio-temporel direct s'exprime :

$$S_\psi = \frac{1}{\sqrt{c_\psi}} \langle \psi | s \rangle \tag{4.4}$$

$$= \frac{1}{\sqrt{c_\psi}} \iint \psi^*(\vec{x},t) s(\vec{x},t) d^2\vec{x} dt \tag{4.5}$$

L'expression de la CWT 2D+t dans le domaine de Fourier (vecteur d'onde/fréquence) donne :

$$S_\psi = \frac{1}{\sqrt{c_\psi}} \langle \hat{\psi} | \hat{s} \rangle \tag{4.6}$$

$$= \frac{1}{\sqrt{c_\psi}} \iint \hat{\psi}^*(\vec{k}, \omega) s(\vec{k}, \omega) d^2\vec{k} d\omega \tag{4.7}$$

A partir de cette expression de la T.O. ST, nous serons en mesure d'établir une famille d'ondelettes plus complète adaptée à un ensemble de paramètres de mouvement $g = \{a, \vec{b}, \tau, c, \theta\}$. Avant de présenter l'adaptation de l'ondelette aux paramètres du mouvement, il est nécessaire de justifier cette adaptation. La section qui suit montre que l'analyse d'un mouvement dans une scène peut se faire par compensation de mouvement puis application d'une transformée en ondelettes, mais aussi sans compensation par application d'une ondelette "adaptée" au mouvement.

4.3 Filtrage compensé en mouvement

L'estimation de mouvement et le filtrage le long des trajectoires d'un mouvement font référence au filtrage par ondelettes compensées en mouvement. Deux approches consistent alors :

1) A "redresser" ("unwarp", voir Fig. 4.1) d'abord le signal (l'image) le long de sa trajectoire puis à y appliquer des ondelettes non-adaptées au mouvement.
2) A *adapter* les ondelettes au mouvement (*motion tuning*) puis à les appliquer au signal (l'image) déformé selon sa trajectoire.

Ces deux approches résultent d'une conséquence directe de l'application d'un opérateur (linéaire ou non) dans l'expression de la convolution entre signal et ondelette. Si l'on traduit le mouvement par une transformation affine (une représentation élémentaire du mouvement) $\mathcal{A} \quad (\mathbb{R}^n \times \mathbb{R} \longrightarrow \mathbb{R}^n \times \mathbb{R})$ appliquée au signal :

$$A \begin{pmatrix} x \\ t \end{pmatrix} \longrightarrow \begin{pmatrix} x' \\ t' \end{pmatrix}$$

l'application de l'opérateur dans la convolution donne le résultat suivant :

$$\int_{\mathbb{R}^n \times \mathbb{R}} \psi \left[\mathcal{A} \begin{pmatrix} x \\ t \end{pmatrix} \right] s \begin{pmatrix} x \\ t \end{pmatrix} d^n x dt \quad = \quad \frac{1}{|det(\mathcal{A})|} \int_{\mathbb{R}^n \times \mathbb{R}} \psi \begin{pmatrix} x' \\ t' \end{pmatrix} s \left[\mathcal{A}^{-1} \begin{pmatrix} x' \\ t' \end{pmatrix} \right] d^n x' dt' \tag{4.8}$$

avec $det(\mathcal{A}) \neq 0$.

Ainsi, toute transformation, linéaire ou non, qui s'exprime ici dans la matrice $\mathcal{A}$, peut, dans l'analyse, *s'appliquer soit au signal, soit à l'ondelette*. Dans le cas d'une application non linéaire, on utilisera le Jacobien, ou déterminant fonctionnel de la transformation (voir Eq. 4.8), plutôt que le

déterminant. La matrice $\mathcal{A}$ se porte sur un objet en mouvement et possède une signification locale, mais on la considère de façon globale afin d'atteindre précisément cet objet en mouvement.

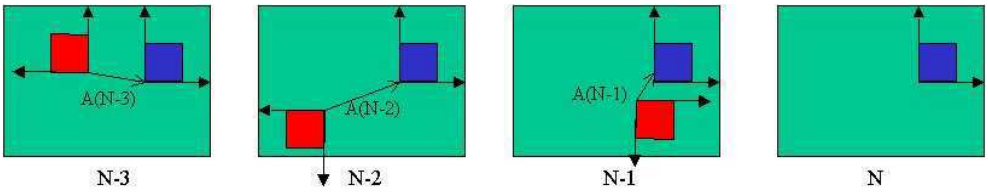


FIG. *4.1 – Opérateur de compensation (unwarping)*

4.4 Opérateurs de transformation applicables à une ondelette

Nous allons maintenant montrer comment ces ondelettes, définies dans le domaine spatial ainsi que dans la direction temporelle, peuvent servir à l'analyse du mouvement. Ceci se fait en plusieurs étapes :
- On montre d'abord que la compensation du mouvement n'est pas opérée sur l'image elle-même, afin de retrouver l'image d'origine et les paramètres de mouvement (problème inverse), mais que c'est l'ondelette adaptée au mouvement qui va analyser et quantifier les paramètres de la transformation subie par l'image ("warped" frame). Ceci est possible grâce à un changement de variable, dans l'expression de la T.O., qui permet de reporter l'action de *l'opérateur de transformation* (ou de "mouvement") du signal vers l'ondelette.
- Dans une deuxième phase nous montrons quelles sont les transformations qui nous intéressent, quelles sont leurs expressions dans les domaines direct et de Fourier et comment elles s'appliquent à une fonction analysatrice.
- Dans une troisième phase nous montrons l'application de certains de ces opérateurs de transformation à une ondelette particulière : l'ondelette de Morlet.

Nous partons de la T.O. redondante que nous avons définie pour un domaine spatio-temporel (et son espace dual dans Fourier). Les paramètres de cette transformée sont $\vec{b}$ pour la translation spatiale, τ pour la translation temporelle et toujours a pour le paramètre d'échelle spatial et temporel (le paramètre est pour l'instant commun aux deux domaines). Nous allons introduire ici d'autres transformations qui peuvent nous permettre d'analyser le signal de façon plus précise et de rendre ce modèle d'ondelette d'analyse plus complexe et plus performant. Ce champ d'étude a déjà été abordé par plusieurs auteurs dont Duval-Destin, F. Mujica, J.P. Leduc.

Remarque 1 *Nous avons montré dans la section précédente l'équivalence entre l'action de l'opérateur $\mathcal{A}$ sur le signal $s(\vec{x},t)$ et sur l'ondelette $\psi(\vec{x},t)$. Dorénavant nous ne nous intéresserons plus qu'à l'action d'opérateurs variés sur l'ondelette uniquement, et non plus sur le signal, afin de construire de nouvelles ondelettes d'analyse, comme par exemple la translation spatio-temporelle $[T^{\vec{b},\tau}\,\psi](\vec{x},t)$ appliquée à l'ondelette d'analyse ψ (voir ci-dessous).*

On considère maintenant une transformation quelconque agissant sur le signal $s(\vec{x},t)$. Cette transformation est représentée par l'opérateur **A** et par un vecteur de paramètres **g**. On note $[\mathcal{A}_g\, s](\vec{x},t)$ l'action de l'opérateur dans le domaine spatio-temporel et $[\hat{A}_g\, \hat{s}](\vec{x},t)$ son action dans le domaine vecteur d'onde-fréquence. Ces transformations sont unitaires, donc satisfont le principe de conservation de l'énergie :

$$|s(\vec{x},t)| = |s(\vec{k},\omega)| = |[A_g\, s](\vec{x},t)| \tag{4.9}$$
$$= |[\hat{A}_g\, \hat{s}](\vec{k},\omega)| \tag{4.10}$$

Nous montrons maintenant comment les transformations qui nous intéressent (translation spatio-temporelle, changement d'échelle, rotation, adaptation à la vitesse) agissent sur l'ondelette d'analyse. Nous verrons plus loin, afin d'associer l'effet des paramètres de la CWT aux transformations du mouvement, comment la CWT redistribue l'énergie dans le domaine vecteur d'onde/fréquence pour un signal donné (section 4.11).
Considérons, dans le domaine 2D+t, un objet soumis aux quatre transformations citées ci-dessus. Ces transformations s'appliquent donc à des ondelettes spatio-temporelles afin de les adapter à l'analyse des mouvements correspondant aux transformations. Dans les figures qui suivent nous utilisons l'ondelette de Morlet, ou plutôt son enveloppe, pour décrire l'effet de chacun des opérateurs sur cette ondelette. L'ondelette de Morlet (voir Fig. 4.7) se représente par une enveloppe gaussienne en 3 dimensions. Nous négligeons dans les représentations ci-dessous les queues des gaussiennes et n'utilisons que l'enveloppe, ellipsoïdale, de l'ondelette ([AMV99]

• **Translation spatio-temporelle**.
Dans cet exemple, l'ondelette est translatée en un point du domaine spatio-temporel. La transformation associée se note $T^{\vec{b},\tau}$ et est définie par :

$$[T^{\vec{b},\tau}\, \psi](\vec{x},t) = \psi(\vec{x}-\vec{b}, t-\tau) \tag{4.11}$$
$$[\hat{T}^{\vec{b},\tau}\, \hat{\psi}](\vec{k},\omega) = e^{-j(\vec{k}.\vec{b}+\omega\tau)}\hat{\psi}(\vec{k},\omega) \tag{4.12}$$

avec $\binom{b}{\tau} \in \mathbb{R}^n \times \mathbb{R}$.

• **Changement d'échelle**.

$$[D^a\, \psi](\vec{x},t) = a^{-3/2}\psi(\frac{\vec{x}}{a}, \frac{t}{a}) \tag{4.13}$$
$$[\hat{D}^a\, \hat{\psi}](\vec{k},\omega) = a^{+3/2}\hat{\psi}(a\vec{k}, a\omega) \tag{4.14}$$

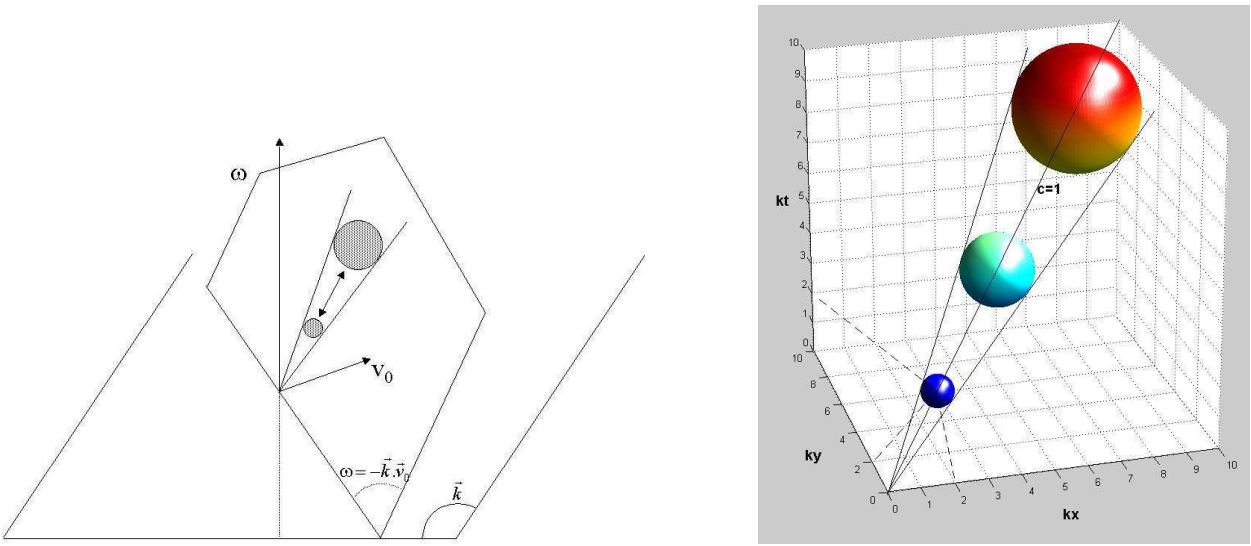


FIG. *4.2 – a) Représentation de l'adaptation de l'ondelette au changement d'échelle a. Les ondelettes restent concentrées dans un cône autour du plan de vitesse constante donné par* $\omega = -k\vec{v}_0$ *b) Représentation 3D de la variation d'échelle sur un filtre adapté à la vitesse* $c = 1$ *et pour des valeurs d'échelle* $a = 1, 1.5, 2$.

• **Adaptation à la rotation**.
La transformation R^θ réalise, dans l'espace direct, une rotation de l'ondelette sur les coordonnées spatiales autour de l'axe des temps :

$$[R^\theta\ \psi](\vec{x}, t) = \psi(r^{-\theta}\vec{x}, t) \tag{4.15}$$
$$[\hat{R}^\theta\ \hat{\psi}](\vec{k}, \omega) = \hat{\psi}(r^{-\theta}\vec{k}, \omega) \tag{4.16}$$

avec

$$r^{+\theta} = \begin{pmatrix} cos\theta & sin\theta \\ -sin\theta & cos\theta \end{pmatrix} \tag{4.17}$$

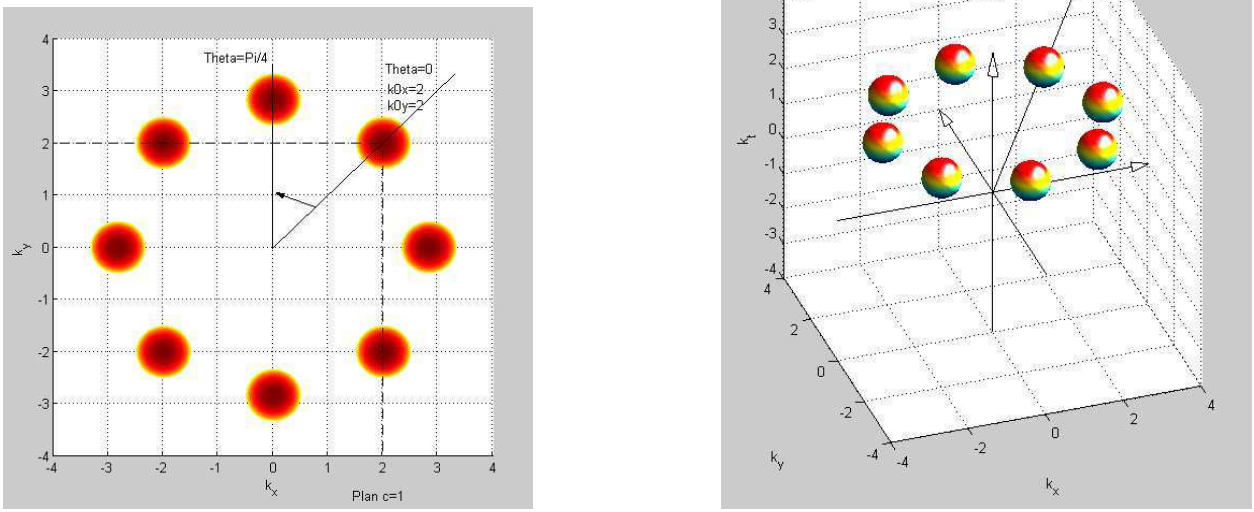


FIG. *4.3 – a) Représentation de l'adaptation de l'ondelette pour des rotations de* 0 *à* 2π *par incrément de* $\pi/4$*. Les ondelettes restent concentrées dans un plan d'inclinaison* $c = 1$ *mais en rotation b) Représentation 3D de la rotation pour des ondelettes sphériques* ($c = 1$) *montrant l'inclinaison constante du plan au cours de la rotation, ce qui permet de modifier l'orientation d'une ondelette adaptée à une vitesse sans modifier cette vitesse.*

• **Adaptation à la vitesse**.
La transformation de vitesse, notée Λ^c, peut être considérée comme deux changements d'échelle effectués indépendamment sur les variables d'espace et de temps. L'adaptation à une vitesse est obtenue simultanément par contraction/dilatation dans l'espace des positions et respectivement dilatation/contraction dans l'espace des temps. Dans la combinaison de ces deux opérations, le volume de l'ondelette est conservé. La transformation s'exprime :

$$[\Lambda^c\ \psi](\vec{x}, t) = \psi(c^{-q}\vec{x}, c^p t) \tag{4.18}$$
$$[\hat{\Lambda}^c\ \hat{\psi}](\vec{k}, \omega) = \psi(c^{+q}\vec{k}, c^{-p}\omega) \tag{4.19}$$

Ainsi considérés, les paramètres p et q sont choisis tels que l'opérateur Λ^c est unitaire et qu'il projette le plan $\vec{v}_0$ dans le plan $c\vec{v}_0$. Ces deux contraintes sont exprimées dans un système à deux équations linéaires ainsi défini :

1) Hypothèse d'unitarité :

$$\begin{aligned} |\psi(\vec{x},t)|^2 &= |[\Lambda^c\ \psi](\vec{x},t)|^2 \\ &= \iint |\psi(c^{-q}\vec{x},c^p t)|^2 d^2\vec{x}dt \\ &= c^{2q-p}\iint |\psi(\vec{x'},t')|^2 d^2\vec{x'}dt' \\ &= c^{2q-p}|\psi(\vec{x},t)|^2 \end{aligned} \tag{4.20}$$

La contrainte d'unitarité impose donc $p = 2q$.
2) Projection du plan de vélocité sur $c\vec{v_0}$ (voir relation 3.3) :

$$\begin{aligned} &c^q\vec{k}.v_0 + c^{-p}\omega = 0 \\ &\omega = -\vec{k}.(c^{p+q}\vec{v}_0) \\ &\omega = -\vec{k}.(c\vec{v}_0) \end{aligned} \tag{4.21}$$

ce qui implique $p + q = 1$.
Le système formé par les contraintes 4.20 et 4.21 conduit donc aux valeurs admissibles :

$$p = +2/3 \tag{4.22}$$
$$q = +1/3 \tag{4.23}$$

et la transformation de vitesse peut finalement être exprimée par :

$$\Lambda^c\ \psi(\vec{x},t) = \psi(c^{-1/3}\vec{x},c^{2/3}t) \tag{4.24}$$
$$\hat{\Lambda}^c\ \hat{\psi}(\vec{k},\omega) = \hat{\psi}(c^{+1/3}\vec{k},c^{-2/3}\omega) \tag{4.25}$$

Signification de l'adaptation à la vitesse
Nous venons de dire que la transformation qui permet l'adaptation de l'ondelette à la vitesse est obtenue simultanément par contraction/dilatation dans le domaine spatial et réciproquement dilatation/contraction dans le domaine temporel. La figure 4.8 montre la contraction d'un facteur deux du support de l'ondelette de Morlet temporelle par rapport à sa version spatiale pour obtenir une vitesse $c = 2$. Les figures 4.9 et 4.10 montrent les versions directe et spectrale de l'ondelette temporelle pour une adaptation à des vitesses $c = 2$ et $c = 10$.
Les figures 4.4, 4.5 et 4.6 montrent, dans des représentations 2D et 3D, comment l'ondelette s'adapte à la vitesse de la même façon que le système psycho-visuel humain. Dans le domaine spectral, l'adaptation de l'ondelette à une vitesse élevée provoque son étirement sur l'axe du vecteur-d'onde temporel et sa contraction sur l'axe spatial. Dans le domaine direct c'est l'inverse. Ce comportement correspond à notre système psycho-visuel. Afin d'être visibles, des motifs en déplacement rapide doivent être grands et des motifs de petite taille doivent être lents. L'augmentation de la résolution temporelle de l'ondelette nécessite sa compression sur l'axe temporel au détriment de la résolution spatiale qui varie en sens inverse.

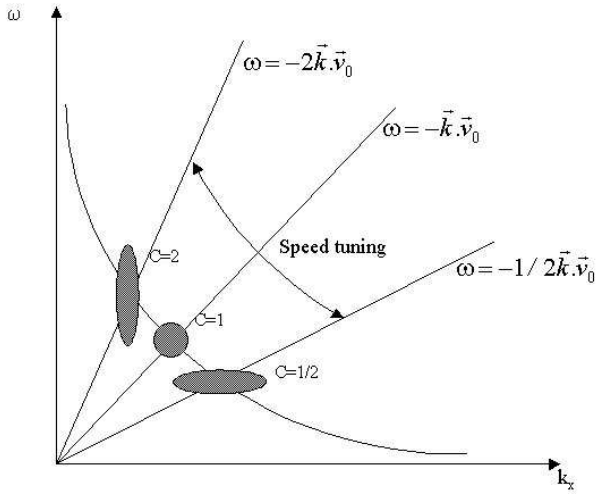


FIG. *4.4 – Adaptation de l'ondelette à la vitesse. Cette représentation dans le domaine spectral est simplifiée à deux dimensions. Cela permet de mettre en évidence comment l'ondelette, de forme circulaire (sphérique en 3D) pour une vitesse $\vec{v}_0 = 1$, est étirée dans la direction du vecteur d'onde temporel pour une vitesse plus élevée ($c = 2$). La nature psycho-visuelle de la transformation fait apparaître une contraction dans la direction du vecteur d'onde spatial ce qui donne la forme elliptique à l'ondelette. Les ondelettes adaptées à plusieurs vitesses se distordent et se déplacent le long d'une hyperbole.*

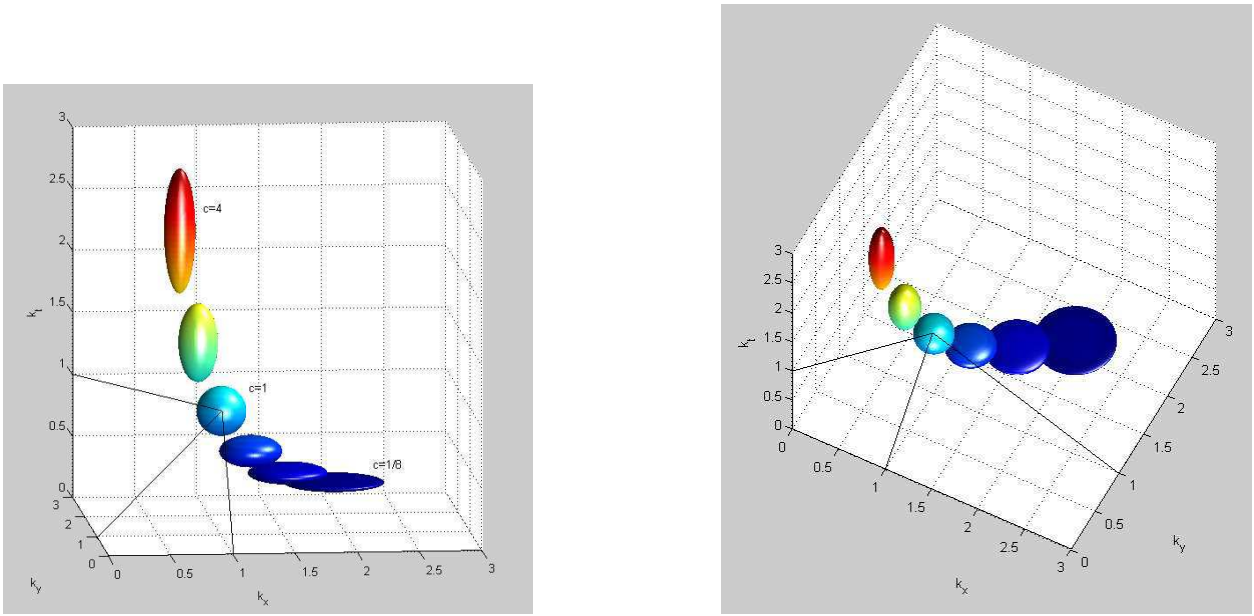


FIG. *4.5 – Adaptation de l'ondelette aux vitesses $c = 4, 2, 1, 1/2, 1/4, 1/8$. a) Tracé des ellipsoïdes montrant la déformation, en fonction du paramètre c de vitesse, de part et d'autre d'une ondelette sphérique (adaptée à la vitesse $c = 1$). b) Tracé 3D des ellipsoïdes le long d'une hyperbole montrant l'effet psycho-visuel de rétrécissement du champ spatial (élargissement pour les vecteurs d'ondes spatiaux) aux basses vitesses.*

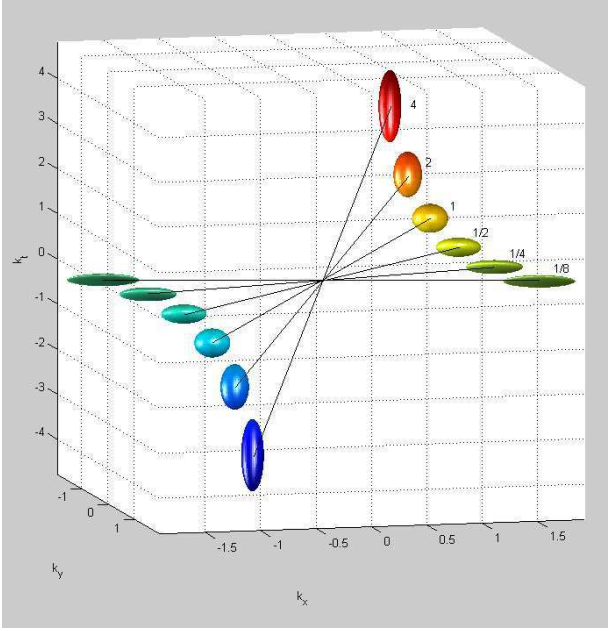


FIG. *4.6 – Adaptation de l'ondelette à la vitesse. Tracé 3D des ellipsoïdes symétriques. Les vitesses sont* $c = 4, 2, 1, 1/2, 1/4, 1/8$

Transformation composite appliquée à une ondelette quelconque

Les opérateurs de transformation décrits précédemment ont permis de définir un espace de paramètres *g étendu* pour l'ondelette, ce qui augmente les capacités d'analyse de cette ondelette au delà des paramètres classiques d'échelle (fréquence) et de la position. Le nouvel espace de paramètre est : $\mathbf{g} = \{a, \vec{b}, \tau, c, \theta\}$.

L'application de l'ensemble des opérateurs aboutit à une **transformation composite** Ω_g formée de : l'homothétie spatiale (paramètre d'échelle), la translation spatiale et temporelle, la rotation et la vélocité. Son application sur une ondelette 1D ψ donne :

$$\boxed{[\Omega_g \psi](\vec{x}, t) = [T^{\vec{b},\tau} R^{\theta} \Lambda^{c} D^{a} \psi](\vec{x}, t)} \tag{4.26}$$

et en remplaçant chaque opérateur de transformation par son expression, on obtient l'expression développée de l'ondelette 1D ψ adaptée aux différentes transformations, dans le *domaine spatio-temporel direct* :

$$\boxed{[\Omega_g \psi](\vec{x}, t) = a^{-3/2} \psi \left(\frac{c^{-1/3}}{a} r^{-\theta}(\vec{x} - \vec{b}), \frac{c^{+2/3}}{a}(t - \tau) \right)} \tag{4.27}$$

qui s'exprime aussi dans le *domaine spatio-temporel de Fourier* :

$$[\Omega_g\psi](\vec{k},\omega) = [T^{\vec{b},\tau}R^{\theta}\Lambda^{c}D^{a}\psi](\vec{k},\omega) \tag{4.28}$$

$$= a^{3/2}\hat{\psi}\left(ac^{+1/3}r^{-\theta}\vec{k}, \frac{1}{a}c^{-2/3}\omega\right)e^{-j(\vec{k}\vec{b}+\omega\tau)} \tag{4.29}$$

En pratique, il sera plus facile de se placer dans le domaine de Fourier pour effectuer les produits terme à terme qui remplaceront les convolutions. De plus l'adoption d'ondelettes séparables DDM permet d'effectuer ces produits successivement en ligne, colonne puis pulsation (vecteur d'onde) temporel $\omega = k_t$. Dans le cas de l'utilisation des constructions d'ondelettes galiléennes de Leduc et al., les CWT se font avec des ondelettes non-séparables.
Nous utiliserons les ondelettes DDM qui possèdent la propriété de séparabilité définie, pour la réponse en fréquence, par :

$$\hat{\psi}_g(\vec{k},\omega) = \hat{\psi}_g(\vec{k}) \times \hat{\psi}_g(\omega) \tag{4.30}$$

à condition que l'ondelette mère possède les mêmes propriétés de séparabilité, c'est-à-dire que la propriété :

$$\hat{\psi}(\vec{k},\omega) = \hat{\psi}(\vec{k}) \times \hat{\psi}(\omega) \tag{4.31}$$

est vérifiée. C'est le cas pour l'ondelette de Morlet que nous présentons dans la section suivante.

4.5 Choix d'une ondelette

Nous nous intéressons maintenant à l'ondelette elle-même. L'ondelette de Morlet est une bonne candidate car elle possède les propriétés de compacité en temps et en fréquence, ce qui offre la possibilité de réaliser les calculs dans l'espace de Fourier en gardant une bonne précision dans l'espace temporel des vitesses. C'est une ondelette à valeurs complexes. La version 1D dans le domaine direct, spatial, s'exprime sous la forme du produit d'une fonction gaussienne (terme évanescent) par une exponentielle complexe de fréquence k_0 (terme oscillant) :

$$\boxed{\psi_{k_0}(x) = e^{-\frac{1}{2}x^2}.e^{ik_0x}} \tag{4.32}$$

et sa version dans le domaine spectral est :

$$\hat{\psi}_{k_0}(k) = e^{\frac{1}{2}(k-k_0)^2} \tag{4.33}$$

Cette ondelette de Morlet constitue l'ondelette mère c'est à dire celle qui engendre les autres ondelettes déclinées par la variation des différents paramètres : position, échelle, vitesse etc. Une transformée en ondelettes réelles est *complète* et *préserve l'énergie* tant que l'ondelette satisfait une *condition d'admissibilité* donnée par le théorème ci-dessous. Ce théorème a été établi d'une part par le mathématicien Calderon en 1964 et de l'autre par Grossman et Morlet. Lorsque ce théorème a été établi par Calderon, celui-ci considérait la T.O. comme une famille d'opérateurs de convolution. C'est exactement le cadre dans lequel nous travaillons, c'est-à-dire non pas celui d'une famille d'opérateurs définissant une base, la transformation utilisée ici n'étant pas une transformée orthogonale, mais celui d'une famille d'opérateurs de convolution.

Théorème 2 (Calderon, Grossman, Morlet) *Une transformée en ondelettes est complète et préserve l'énergie si elle satisfait la condition d'admissibilité définie par :*

$$c_\psi = 2\pi^3 \int_{\mathbb{R}^2} \int_{\mathbb{R}} \frac{|\hat{\psi}(\vec{k},\omega)|^2}{|\vec{k}|^2|\omega|} d^2\vec{k}d\omega < \infty \tag{4.34}$$

Ainsi pour être considérée comme *admissible*, une ondelette mère doit répondre à la condition de carré intégrable, ce qui revient à avoir une énergie finie. La condition de carré intégrable revient à la convergence de l'intégrale vers la limite c_ψ. Première remarque sur cette condition d'admissibilité : elle implique que $\hat{\psi}(0) = 0$. Deuxième remarque : si $\hat{\psi}$ est continûment différentiable, la condition d'admissibilité est vérifiée.
La complétude signifie que toute fonction peut être décomposée sous la forme d'une somme de produits scalaires entre la fonction et l'ondelette mère dilatée , translatée, mais aussi soumise aux autres paramètres qui "l'adaptent" : rotation, vitesse.

Si l'ondelette satisfait la condition d'admissibilité, la transformée en ondelettes spatio-temporelle peut alors s'exprimer, dans le domaine vecteur-d'onde/fréquence $(\vec{k},\omega)$, par :

$$S_{a,b,\tau,\theta,c} = \frac{1}{\sqrt{c_\psi}} < \hat{\psi}_{(a,b,\tau,\theta,c)} | \hat{s} > \tag{4.35}$$

• La version en trois dimensions (2D+T) de cette ondelette de Morlet, modulée dans les directions $\vec{x} = (x_1, x_2)$ et t, nous donne la *version spatio-temporelle* de l'ondelette de Morlet classique (Fig. 4.7). Elle s'exprime sous la forme de gaussiennes modulées selon les variables d'espace $\vec{x}$ et de temps t. Par rapport à l'expression initiale 1D, un terme d'admissibilité (elle doit appartenir à l'espace L^2 des fonctions de carré intégrable, donc convergent) par rapport à chacune des variables x et t a été ajouté :

$$\psi(\vec{x},t) = \underbrace{(e^{-1/2|\vec{x}|^2}.e^{ik_0\vec{x}} - \underbrace{e^{-1/2(|\vec{x}|^2+|\vec{k_0}|^2)}}_{admiss})}_{spatial} \times \underbrace{(e^{-1/2t^2}.e^{i\omega_0 t} - \underbrace{e^{-1/2(t^2+{\omega_0}^2)}}_{admiss})}_{temporel} \tag{4.36}$$

que l'on peut reécrire :

$$\psi(\vec{x},t) = \underbrace{e^{-\frac{\vec{x}^2+t^2}{2}}.e^{-i(k_0\vec{x}+\omega_0 t)}}_{A} - \underbrace{e^{-\frac{\vec{x}^2+t^2}{2}}.e^{-\frac{{k_0}^2+{\omega_0}^2}{2}}}_{B} \tag{4.37}$$

Cette dernière expression se limite au terme A si le terme d'admissibilité (B) est négligeable. En l'occurrence, ceci se présente pour $|k_0|$ et $\omega_0 \geq \pi\sqrt{\frac{2}{Ln2}} \simeq 5.336$ [DDM93, AM96].
Dans la suite on adopte cette même valeur pour $|k_0|$ et ω_0, ce qui conduit à une vitesse : $v_0 = \frac{\omega}{k} = 1(pix/fr)$.
On obtient alors l'expression de *l'ondelette de Morlet spatio-temporelle simplifiée*, dans l'espace direct, par annulation du terme d'admissibilité :

$$\boxed{\psi(\vec{x},t) = (e^{-1/2|\vec{x}|^2}.e^{ik_0x}) \times (e^{-1/2t^2}.e^{i\omega_0 t})} \tag{4.38}$$

• La version duale, dans le domaine de Fourier, de l'ondelette de Morlet spatio-temporelle décrite par 4.36 s'exprime :

$$\hat{\psi}(\vec{k},\omega) = (e^{-\frac{1}{2}|\vec{k}-\vec{k}_0|^2} - e^{-\frac{1}{2}(|\vec{k}|^2+|\vec{k}_0|^2)}) \times (e^{-\frac{1}{2}(\omega-\omega_0)^2} - e^{-\frac{1}{2}(\omega^2+\omega_0^2)}) \tag{4.39}$$

Cette relation, lorsqu'on annule aussi les termes d'admissibilité, donne l'expression *simplifiée* de l'ondelette ST (2D+T) de Morlet dans le domaine de Fourier :

$$\boxed{\hat{\psi}(\vec{k},\omega) = (e^{-\frac{1}{2}|\vec{k}-\vec{k}_0|^2}) \times (e^{-\frac{1}{2}(\omega-\omega_0)^2})} \tag{4.40}$$

L'utilisation de l'ondelette de Morlet est motivée par les propriétés de *localisation (ou compacité) optimale* en temps et en fréquence des fonctions gaussiennes. Cependant, nous le verrons, le choix d'une ondelette moins oscillante est préférable dans des applications qui ne s'intéressent pas au caractère ondulatoire du phénomène mais plutôt à sa localisation spatio-temporelle.
L'ondelette de Morlet est choisie comme une ondelette mère particulière. Elle autorise les filtres passe-bande de forme gaussienne admissibles comme ondelettes de Galilée. L'ondelette statique $v = 0$ est dessinée comme un filtre non-séparable pour éviter au filtre son annulation sur le plan $\omega = 0$. Elle est ensuite déformée comme un ballon aplati sur le plan des fréquences spatiales $\omega = 0$. La position de l'ondelette statique est ensuite déterminée par le vecteur de polarité (ou positionnement) k_0 afin d'être positionnée avant l'application des transformations de vélocité.

4.6 Transformation composite appliquée à l'ondelette de Morlet

Pour obtenir une ondelette de 2D+T adaptée à l'ensemble des paramètres $g = \{a, \vec{b}, \tau, \theta, c\}$, nous appliquons aux variables d'espace et de temps modifiées par la transformation composite 4.27 dans l'expression de l'ondelette de Morlet spatio-temporelle simplifiée 4.38 que nous venons de décrire. Nous obtenons une ondelette :
- de Morlet simplifiée (termes d'admissibilité négligeables grâce aux valeurs de k_0 et ω_0 adoptées) qui possède des propriétés de compacité à la fois dans l'espace direct et dans l'espace réciproque.
- Développée pour une analyse spatio-temporelle (2D+T) et opérable dans le domaine direct ou fréquentiel.
- Adaptée à un ensemble de paramètres de mouvement $g = \{a, \vec{b}, \tau, \theta, c\}$
L'application de la transformation composite 4.27 à l'ondelette de Morlet 4.38 simplifiée donne l'expression dans le domaine ST direct :

$$\boxed{\psi_{(a,\vec{b},\tau,\theta,c)}(\vec{x},t) = a^{-3/2}.\underbrace{e^{-\frac{c^{-2/3}}{2a^2}|\vec{x}-\vec{b}|^2} \times e^{-i\frac{c^{-1/3}}{a}\vec{k}_0 r^\theta(\vec{x}-\vec{b})}}_{terme\ spatial}.\underbrace{e^{-\frac{c^{4/3}}{2a^2}(t-\tau)^2}.e^{-i\frac{c^{2/3}}{a}\omega_0(t-\tau)}}_{terme\ temporel}} \tag{4.41}$$

et sa version dans le domaine de Fourier, en prenant 4.28 + 4.40 :

$$\hat{\psi}_{(a,\vec{b},\tau,\theta,c)}(\vec{k},\omega) = \underbrace{a^{3/2}(e^{-\frac{a^2}{2}c^{+2/3}r^{\theta}(\vec{k}-k_0)^2}).\,(e^{-i((\vec{k}-k_0)\vec{b})})}_{terme\ spatial}.\ \underbrace{(e^{-\frac{a^2}{2}c^{-4/3}(\omega-\omega_0)^2}) \times\ (e^{-i((\omega-\omega_0)\tau)})}_{terme\ temporel} \tag{4.42}$$

que l'on peut reécrire sous la forme :

$$\hat{\psi}_{(a,\vec{b},\tau,\theta,c)}(\vec{k},\omega) = a^{3/2}(e^{-\frac{a^2}{2}c^{+2/3}r^{\theta}(\vec{k}-k_0)^2}) \times (e^{-\frac{a^2}{2}c^{-4/3}(\omega-\omega_0)^2}) \times (e^{-i((\vec{k}-k_0)\vec{b}+(\omega-\omega_0)\tau)}) \tag{4.43}$$

La figure ci-dessous donne deux représentations (2D et 3D) de l'ondelette de Morlet non-séparable spatio-temporelle. Nous montrons aussi sur la figure de gauche comment l'effet du paramètre d'anisotropie agit sur la forme de l'ondelette en l'aplatissant comme un disque dans le plan de la vitesse sélectionnée, ce qui la rend plus sélective à cette vitesse (voir section 4.7 sur la sélectivité). Comme

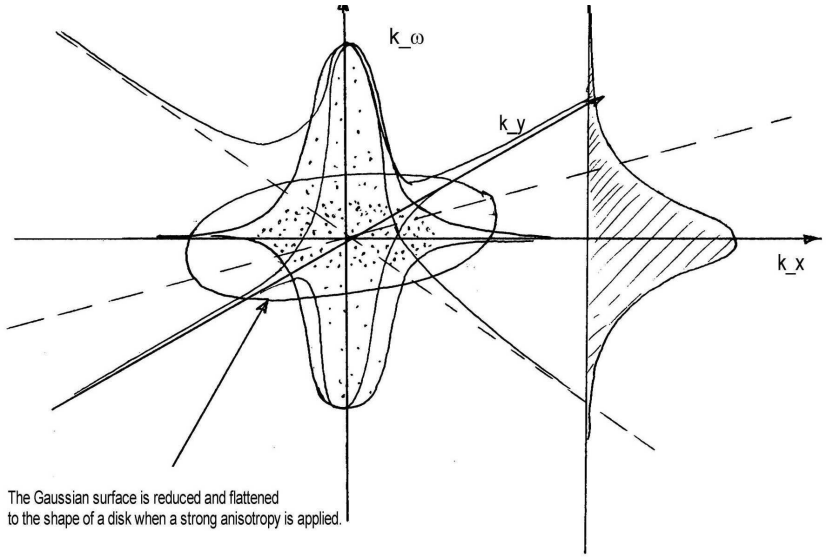


FIG. *4.7 – L'ondelette de Morlet spatio-temporelle non-séparable peut être représentée par une surface gaussienne 2D spatiale (à gauche). On obtient sa représentation non-séparable 3D en faisant varier la densité de points de cette surface, selon l'axe vertical ω, par une gaussienne fréquentielle (ou temporelle), représentée à droite. En faisant l'approximation de négliger les queues gaussiennes, l'ondelette prend alors la forme de l'ellipsoïde rencontrée sur la figure 4.4. Lorsque l'ondelette est construite avec une forte anisotropie temporelle ϵ_t, elle est assimilable à un disque (centre de la figure de gauche).*

nous travaillons avec la version séparable de ce filtre 3D, chaque filtre peut être représenté comme une seule ondelette de Morlet. L'adaptation à la vitesse, comme nous l'avons dit, est réalisée par

contraction ou dilatation dans l'espace spatial, respectivement dilatation ou contraction dans l'espace temporel. Ceci est mis en évidence sur les figures ci-dessous, où l'ondelette est adaptée à une vélocité $c = 2$. La pulsation de l'ondelette est choisie avec $k_0 = \omega_0 = 6$ ce qui, nous l'avons vu, permet de négliger le terme d'admissibilité dans le modèle de l'ondelette.

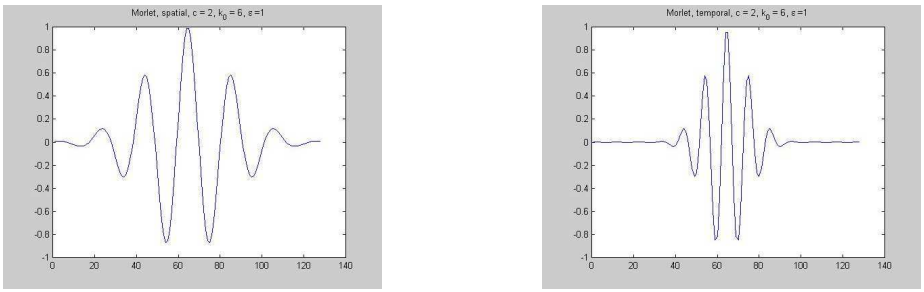


FIG. *4.8 – Adaptation de l'ondelette de Morlet séparable spatio-temporelle à une vitesse* $c = 2$. *Affichage dans le domaine direct a) Ondelette de Morlet sur l'axe spatial b) Ondelette sur l'axe temporel (facteur de contraction de 2 permettant l'adaptation à la vitesse de* 2 *pixels/frame)*

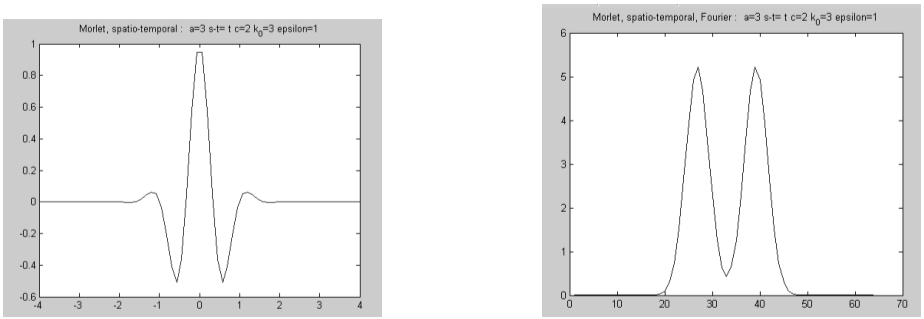


FIG. *4.9 – Ondelette de Morlet adaptée à la vitesse : a) A gauche, ondelette de Morlet temporelle séparable adaptée à une vitesse* $c = 2$ *b) A droite, sa version spectrale et le décalage du spectre par rapport à la fréquence nulle* $\omega = 0$

4.7 Sélectivité de l'ondelette

• La sélectivité de l'ondelette est un point important dans cette approche. Elle permet de rendre cette ondelette plus ou moins sélective autour d'une vitesse précise. Le but est de permettre au filtre de capter une "gamme" plus ou moins large de vitesses. Cette sélectivité est obtenue par

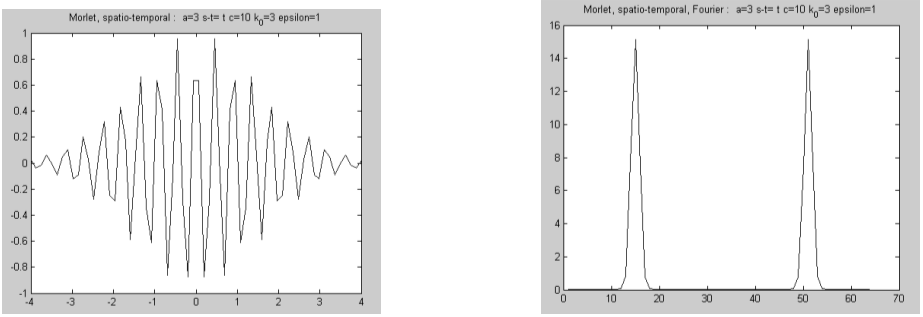


FIG. *4.10 – Ondelette de Morlet adaptée à la vitesse : a) A gauche, ondelette de Morlet temporelle séparable adaptée à une vitesse $c = 10$ b) A droite sa version spectrale qui montre la translation du spectre dans un plan de fréquence, donc de vitesse, plus élevé.*

modification de la variance de l'ondelette au moyen d'un paramètre d'anisotropie ϵ appliqué aux filtres spatiaux (ou temporels). Le paramètre ϵ contrôle la variance de l'ondelette par rapport au plan de référence de la vélocité à laquelle est adapté le filtre.
L'application est réalisée en remplaçant $\vec{x}$ et $\vec{k}$ par $A^{-1}\vec{x}$ et $A\vec{k}$ respectivement, avec :

$$A = \begin{bmatrix} 1 & 0 \\ 0 & \epsilon \end{bmatrix} \qquad (et : A^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & 1/\epsilon \end{bmatrix}) \tag{4.44}$$

• Afin d'obtenir une forte sélectivité en vélocité, une forte anisotropie spatio-temporelle ϵ_t (qui caractérise l'aplatissement de l'ondelette) est aussi nécessaire pour une analyse en ondelettes précise. La version spatio-temporelle directe de l'ondelette, incorporant la matrice de sélectivité C, s'écrit, en posant $X = (\vec{x}, t)$:

$$\begin{aligned} \psi(\vec{x}, t) &= \psi(X) \\ &= e^{-i<k_0|X>} e^{-\frac{1}{2}<X|CX>} - \underbrace{e^{-\frac{1}{2}<k_0|Ck_0>} e^{-\frac{1}{2}<X|CX>}}_{admiss.} \\ &\simeq e^{ik_0X} e^{-\frac{1}{2}CX^2} \end{aligned} \tag{4.45}$$

avec :

$$C = \begin{pmatrix} 1/\epsilon_x & 0 & 0 \\ 0 & 1/\epsilon_y & 0 \\ 0 & 0 & 1/\epsilon_t \end{pmatrix} \tag{4.46}$$

L'expression 4.45 ci-dessus montre que la matrice diagonale C agit comme $\frac{1}{\sigma^2}\mathbf{I}$ dans le terme qui représente l'enveloppe gaussienne de l'ondelette. Autrement dit, ses termes représentent les inverses des variances des ondelettes séparables, et ϵ représente la variance des ondelettes.
La version spatio-temporelle spectrale, incluant la matrice de sélectivité D, s'écrit, en posant

$K = (\vec{k}, \omega)$:

$$\begin{aligned}\hat{\psi}(\vec{k},\omega) &= \hat{\psi}(K) \\ &= |det(D)|^{\frac{1}{2}} \ (e^{-\frac{1}{2}<(K-k_0)|D(K-k_0)>} - \underbrace{e^{-\frac{1}{2}<k_0|Dk_0>} e^{-\frac{1}{2}<K|DK>}}_{admiss.}) \\ &\simeq |det(D)|^{\frac{1}{2}} \ e^{-\frac{1}{2}D(K-k_0)^2} \end{aligned} \tag{4.47}$$

avec :

$$D = C^{-1} = \begin{pmatrix} \epsilon_x & 0 & 0 \\ 0 & \epsilon_y & 0 \\ 0 & 0 & \epsilon_t \end{pmatrix} \tag{4.48}$$

Dans l'expression spectrale, le paramètre D agit comme $\sigma^2\mathbf{I}$ dans le terme qui représente l'enveloppe gaussienne de l'ondelette. D représente donc la matrice diagonale des variances des trois ondelettes dans le domaine de Fourier.

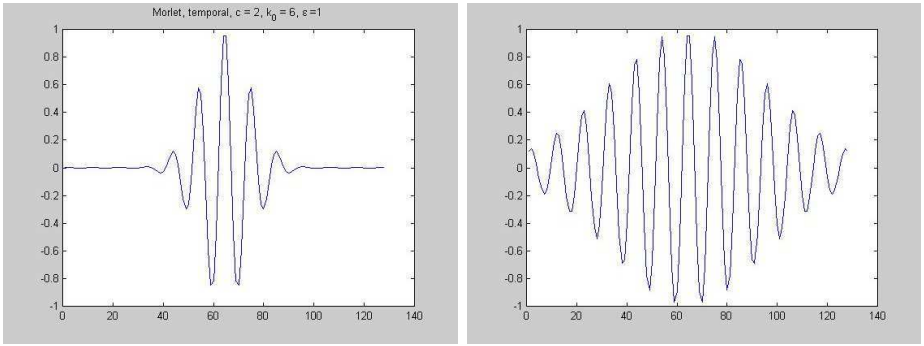


FIG. *4.11 – Variation de la sélectivité de l'ondelette par le paramètre ϵ d'anisotropie a) Ondelette de Morlet sur l'axe temporel, pour c=2 et une anisotropie $\epsilon = 1$ b) Ondelette de Morlet sur l'axe temporel, pour c=2 et une anisotropie $\epsilon = 10$*

Lorsqu'on augmente le paramètre d'anisotropie à $\epsilon = 10$ ou $\epsilon = 100$ (Fig. 4.11), l'adaptation à une vitesse déterminée est plus sélective. En revanche, selon le principe de Heisenberg, la position spatiale de l'objet détecté devient plus approximative. Ce comportement est proche du comportement psycho-visuel humain et montre que ce type d'ondelette est bien adapté à la détection du mouvement dans des approches de vision animée.

4.8 Relation complète pour les trois ondelettes adaptées et la TO correspondante (MTSTWT)

Nous pouvons maintenant donner l'expression des trois ondelettes spatio-temporelles séparables de Morlet dans leur version simplifiée (terme d'admissibilité négligé), pour une adaptation à l'ensemble

des transformations donné par les paramètres $g = a, \vec{b}, \tau, c, \theta$ et avec un paramètre de sélectivité ϵ apporté à la vitesse. Nous donnons les versions spectrales des relations car ce sont aussi celles avec lesquelles nous avons effectué nos simulations.
Puisque nous avons adopté la simplification d'utiliser la même sélectivité pour les deux filtres spatiaux 1D, nous avons les deux premières relations pour les ondelettes spatiales $\hat{\psi}(k_x)$ et $\hat{\psi}(k_y)$:

$$\hat{\psi}_{(a,b_x,\theta,c,\epsilon_x)}(k_x) = \sqrt{Det(D)}a^{3/2}(e^{-\frac{a^2}{2}c^{+2/3}r^{\theta}(k_x-k_0)^2\epsilon_x}) \times \ (e^{-i(|k_x-k_0|b_x)}) \tag{4.49}$$

$$\hat{\psi}_{(a,b_y,\theta,c,\epsilon_y)}(k_y) = \sqrt{Det(D)}a^{3/2}(e^{-\frac{a^2}{2}c^{+2/3}r^{\theta}(k_y-k_0)^2\epsilon_y}) \times \ (e^{-i(|k_y-k_0|b_y)}) \tag{4.50}$$

et pour l'ondelette temporelle $\hat{\psi}(\omega)$:

$$\hat{\psi}_{(a,\tau,c,\epsilon_t)}(\omega) = \sqrt{Det(D)}a^{3/2}(e^{-\frac{a^2}{2}c^{-4/3}(\omega-\omega_0)^2\epsilon_t}) \times \ (e^{-i(\omega-\omega_0)\tau}) \tag{4.51}$$

La transformée "MTSTWT" (Motion-tuned Spatio-temporal Wavelet Transform) associée à ces trois ondelettes s'exprime dans le domaine direct par :

$$W\psi\{g\}s(\vec{x},t) = \int_{\mathbb{R}^2}\int_{\mathbb{R}} s(\vec{x},t)\psi_{a,c,\theta}(\vec{x}-\vec{b},t-\tau)d^2\vec{x}dt \tag{4.52}$$

La MTSTWT s'exprime dans le domaine fréquentiel, en séparant les trois ondelettes :

$$\boxed{\hat{W}\hat{\psi}\{g\}\hat{s}(\vec{k},\omega) = \hat{s}(\vec{k},\omega) \times \hat{\psi}_{\{a,bx,\theta,c\}}(k_x) \times \hat{\psi}_{\{a,by,\theta,c\}}(k_y) \times \hat{\psi}_{\{a,\tau,c\}}(\omega)} \tag{4.53}$$

4.9 Algorithme de traitement par MTSTWT d'une séquence

L'algorithme que nous avons adopté pour le traitement d'une séquence par la TO spatio-temporelle adaptée (la MTSTWT) est décrit par les étapes suivantes :
1) On construit l'ondelette 2D+T dans sa version séparable et spectrale, c'est-à-dire que l'on utilise trois ondelettes ψ_{kx}, ψ_{ky} et ψ_{kt} pour chaque direction dans l'espace de Fourier (ou espace des vecteurs d'onde). Ces ondelettes sont adaptées aux transformations qui nous intéressent, en utilisant l'ensemble des paramètres g, dans les relations 4.49, 4.49 et 4.51. Ces relations incluent la prise en compte d'un paramètre supplémentaire ϵ de sélectivité en vitesse de façon à ajuster la gamme de vitesses qu'elles peuvent détecter.
2) On effectue une transformée de Fourier 3D sur un bloc de la séquence vidéo (par exemple 8 trames successives).
3) On effectue, dans Fourier, le produit de la séquence par chacune des ondelettes séparément dans les directions kx, ky et kt.
4) On effectue une Transformée de Fourier inverse 3D sur le résultat obtenu. Ceci revient à une analyse en ondelette classique mais les coefficients obtenus peuvent fournir, suivant les paramètres assignés aux trois ondelettes, la position du signal par rapport à l'ondelette, son échelle, sa vitesse ou sa rotation.

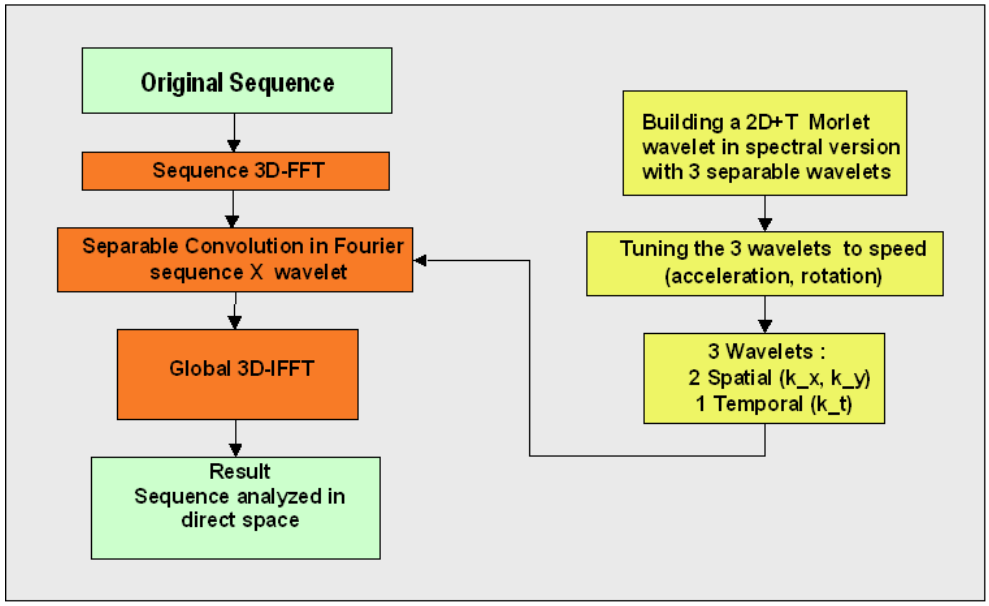


FIG. *4.12 – Synoptique de la transformée en ondelette spatio-temporelle adaptée à la vitesse ($g = a, b, \tau, c$) et éventuellement à l'accélération et la rotation. Cette décomposition a été utilisée dans les analyses par GOF présentées plus loin dans les résultats.*

4.10 Autres familles d'ondelettes associées aux groupes de transformation

Plusieurs famille d'ondelettes ont été synthétisées pour s'adapter à différents groupes de paramètres. L'adaptation des ondelettes à la rotation a été réalisée très tôt notamment par R. Murenzi []. . Ces ondelettes sont construites sur une grille à variation fine de l'angle de rotation. Une autre famille d'ondelettes a été synthétisée pour s'adapter à quatre paramètres dont surtout la vitesse : ce sont les ondelettes galiléennes, développées par J.P. Antoine et I. Mahara [AM98]. L'ensemble des paramètres de cette famille est $g = \{a, (\vec{b}, \tau), \theta, c\}$ où a est l'échelle, $(\vec{b}, \tau)$ la translation (ou position) spatio-temporelle, θ l'orientation spatiale et c la vitesse.
Cette famille galiléenne se rattache à la théorie des ondelettes discrètes *compensées en mouvement*. C'est cet aspect qui nous intéresse ici pour l'estimation de mouvement. La différence majeure entre les espaces multi-dimensionnels homogènes et les espaces spatio-temporels est la présence de mouvements qui déploient le signal le long de trajectoires. La modélisation du mouvement à l'aide de transformations affines bi-dimensionnelles se généralise alors dans le domaine spatio-temporel pour décrire les mouvements de la mécanique et apporter des informations sur la cinématique des scènes vidéo.
La décomposition en opérateurs élémentaires conduit à développer des groupes de transformations

et à exploiter la théorie des représentations de groupe. La construction d'ondelettes continues spatio-temporelles dans les espaces $\mathbb{R} \times \mathbb{R}$ est ensuite traitée par des techniques classiques de calcul. Les familles développées pour l'adaptation à des paramètres multiples sont nombreuses. Citons [DDM93, AM98] :

– Les ondelettes cinématiques ou euclidiennes. C'est le cas le plus simple. Le groupe de transformation G consiste en translations temps-espace et dilatations temps-espace (a, a_0), en rotations (réflexions en 1D) et réflexions dans le temps.
– Les ondelettes de Galilée. G est ici le groupe affine de Galilée, qui combine le groupe (étendu) de Galilée avec des dilatations (a, a_0) de l'espace et du temps *indépendantes*.
– Les ondelettes de Schrödinger.
– Les ondelettes relativistes. G est alors le groupe de Weyl-Poincaré ([AM98], Introduction : space-time wavelets).

D'autres familles ont été développées pour les cas particuliers d'analyse de mouvements affines et la paramétrisation cinématique dans des séquences d'images. Ce sont [Led97] :

– Les ondelettes accélérées.
– Les ondelettes rotationnelles.
– Les ondelettes adaptées à la déformation

Une transformation de déformation peut aussi être réalisée au moyen de plusieurs transformées en ondelettes basées mouvement [CLK99] : les ondelettes de Galilée, les ondelettes "accélérées", représentations d'un autre groupe de Lie, et les ondelettes "sur la variété". Les paramètres sont définis pour ces trois familles par :

4.10.1 Ondelettes de Galilée

Le groupe de Galilée décrit des transformations basées sur la vélocité. Il est défini par l'ensemble de paramètres qui suit :

$$g = \{g \in G | g = (\Phi, \vec{b}, \tau, \vec{v}, a, R)\} \tag{4.54}$$

où Φ est le *paramètre d'extension centrale* du groupe, $\vec{b}$ et τ sont les paramètres de translation spatiale et temporelle, $\vec{v}$ est la vélocité, a est l'échelle et $\mathrm{R} \in SO(n)$ est l'orientation spatiale. La transformée galiléenne s'écrit :

$$W[s(x,t); b, \tau; v, a, r] = c_\Psi^{1/2} < \Psi_{b,\tau;v,a,r} | s > \tag{4.55}$$

$$= c_\Psi^{1/2} \int_{\mathbb{B}R^n \times \mathbb{R}} \Psi \left[r^{-1} \left(\frac{x - b - v}{a} \right), t - \tau \right] s(x,t)\, d^n x\, dt \tag{4.56}$$

4.10.2 Ondelettes accélérées

Le groupe de paramètre g contient maintenant un terme d'adaptation γ à l'accélération :

$$g = \{g \in G | g = (\Phi_2, \Phi_3, \vec{b}, \tau_2, \vec{\gamma}, a, R)\} \tag{4.57}$$

Cette étude permet d'estimer et d'analyser le mouvement dans des séquences d'images numériques en utilisant des expansions selon cette nouvelle transformée avec adaptation à l'accélération. Les

ondelettes peuvent être alors paramétrées par : la translation spatiale et temporelle, la vélocité , l'accélération, le changement d'échelle et la rotation spatiale. Il est alors possible de réaliser des reconstructions sélectives sur les objets accélérés [LCK+98].

4.10.3 Ondelettes sur la variété

$$g = \{g \in G | g = (\Phi, \vec{p}, \tau, \vec{v}, A, R_0)\} \tag{4.58}$$

Ici $\vec{p}$ est un paramètre généralisé de translation spatiale sur la variété. Un autre modèle de transformation de déformation (TD) a été déjà développé dans []. Ces transformations de déformation peuvent être utilisées pour l'analyse du mouvement dans des espaces de dimension supérieure. Prenons l'exemple de capteurs plans : le composant de la vélocité, orthogonal au plan de projection du capteur, n'est pas vu comme une vélocité mais comme une TD impliquant un changement d'échelle. La TD permet alors d'analyser ce mouvement en 3 dimensions.

4.10.4 Classement des familles d'ondelettes adaptées au mouvement

Par analogie avec le développement en séries et la transformée de Fourier, quatre versions différentes d'ondelettes peuvent être étudiées lorsqu'on considère les espaces continus et discrets. Ces versions sont appelées :

- La transformée en ondelettes continue.
- Les séries d'ondelettes ou frames d'ondelettes.
- Les séries d'ondelettes discrètes.
- La transformée en ondelettes discrète.

Le tableau 4.13 résume l'ensemble des transformées en ondelettes spatio-temporelles, leur groupes d'appartenance (série d'ondelettes ou transformée) et leur domaine (continu ou discret). Partant des groupes définis par des ensembles de paramètres spécifiques aux analyses visées, on aboutit à la construction d'ondelettes compensées en mouvement orientées estimation de mouvement ou segmentation.

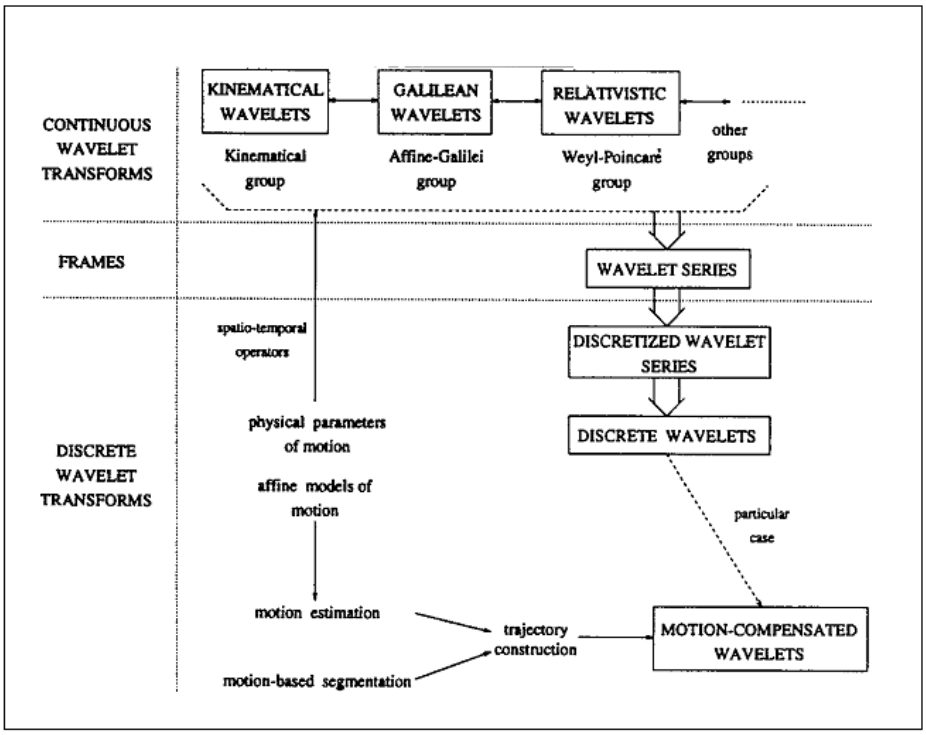


FIG. *4.13 – Tableau résumant les groupes de transformées en ondelettes spatio-temporelles à paramètres multiples et leur développement en ondelettes compensées en mouvement (d'après [LMMS00]).*

4.11 Exemple d'algorithme de poursuite par MTSTWT

Nous montrons ici une méthode d'estimation de mouvement itérative basée sur la maximisation des densités d'énergie associées aux paramètres de mouvement [MLMS00]. Les paramètres de mouvement considérés sont : l'échelle, la position spatio-temporelle (3D), la rotation et la vélocité. Ceci donne lieu à un espace de paramètres à six dimensions, le domaine ondelette. La nature multidimensionnelle de la représentation en ondelettes est à la base de l'algorithme d'estimation de mouvement. La CWT permet la définition d'un ensemble de trois densités d'énergie qui peuvent être utilisées comme fonctions de coût pour l'estimation des paramètres de mouvement.

La trajectoire d'un objet animé dans une scène peut se représenter par une expansion de Taylor en chaque point $(\vec{x}_0, \tau)$:

$$\boxed{\vec{x}(t=\tau) = \vec{x}_0 + \vec{v}\tau + \vec{\gamma}_0 \frac{\tau^2}{2!} + \sum_{i=1}^{\infty} \vec{\gamma}_i \frac{\tau^{i+2}}{(i+2)!}} \tag{4.59}$$

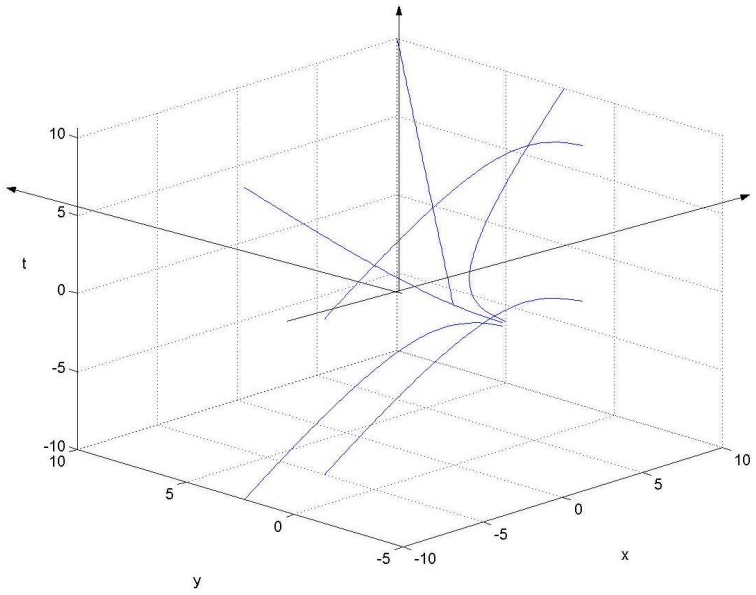


FIG. *4.14 – Trajectoires spatio-temporelles*

4.11.1 Densités d'énergie

Les trois densités d'énergie décrites ci-dessous permettent de mettre à jour l'ensemble des paramètres g au cours du suivi d'un objet dans un suivi itératif trame après trame. La maximisation de chacune des énergies par rapport aux paramètres de mouvement g permet, à chaque itération complète de l'algorithme, de remettre à jour les paramètres de mouvement. Le suivi des paramètres trame par trame permet, dans cette réalisation, de suivre la trajectoire de l'objet et d'interpoler celle-ci en cas d'occlusion. L'algorithme permet donc un filtrage de la trajectoire et offre un suivi "cohérent" de l'objet que ne permettent pas les méthodes de block-matching.

• **Densité d'énergie orientation-vitesse** :
L'intégration est réalisée pour une translation spatiale $\vec{b} = (b_x, b_y)$ sur une région :

$$\mathcal{B} : b_{x_{min}} < b_x < b_{x_{max}} \bigcap b_{y_{min}} < b_y < b_{y_{max}} \tag{4.60}$$

avec une échelle a et une translation temporelle τ fixées.
On obtient alors une densité d'énergie :

$$\mathcal{E}^I_{a_0,\tau_0}(c,\theta) = \int_{\vec{b}\in\mathcal{B}} |\langle \psi_{a_0,c,\theta,\vec{b},\tau_0} \mid s\rangle|^2 \, d^2\vec{b} \tag{4.61}$$

que l'on peut interpréter comme un estimateur local de la vélocité.

• **Densité d'énergie spatiale** :

$$\mathcal{E}^{II}_{a_0,c_0,\theta_0,\tau_0}(\vec{b}) = \frac{1}{{a_0}^4}|\langle\psi_{a_0,c_0,\theta_0,\vec{b},\tau_0} \mid s\rangle|^2 \tag{4.62}$$

- **Densité d'énergie en échelle** :

$$\mathcal{E}^{III}_{c_0,\theta_0,\tau_0}(a) = \frac{1}{a^4}\int_{\vec{b}\in\mathcal{B}} |\langle\psi_{a,c_0,\theta_0,\vec{b},\tau_0} \mid s\rangle|^2 \; d^2\vec{b} \tag{4.63}$$

4.11.2 Description de l'algorithme

Cet algorithme est un exemple d'utilisation de la MTSTWT, ou CWT spatio-temporelle adaptée au mouvement, dans une application de tracking [MLMS00]. Brièvement décrit, l'algorithme est basé sur le calcul des trois densités d'énergie décrites dans la section précédente. Le calcul de la MTSTWT est réalisé dans le domaine spectral, puis les transformées de Fourier inverses 1D temporelle, et 2D spatiale, permettent de revenir à la valeur de l'énergie dans le domaine spatio-temporel direct 4.15. Un algorithme de maximisation (Nelder-Mead) de la densité d'énergie en vitesse-orientation permet d'évaluer les vitesses, dans un voisinage $\vec{b}, \tau$ du GOF et à échelle fixée, en fonction des paramètres de vitesse et d'orientation attribués aux filtres d'analyse au début de l'itération . Cet algorithme est aussi utilisé pour maximiser la densité d'énergie en échelle et mettre à jour le paramètre d'échelle. La mise à jour des paramètres de position et d'échelle est faite séquentiellement avec l'étage de calcul de vitesse.

Une itération de l'algorithme de mise à jour de l'étage de correction de la vélocité

On suppose que cette itération est effectuée entre les instants $\tau_0 = t_i$ et $\tau = t_{i+1}$. L'itération consiste à effectuer une mise à jour de la vélocité mesurée dans un voisinage spatial $\vec{b}$ et pour une valeur l'échelle a_0 et de la position temporelle τ

1. On effectue une DFT 3D sur un GOF de taille $M \times N \times K$.
2. On évalue $\hat{\bar{\psi}}_{a_0,c_0,\theta_0}(\vec{k},\omega)$:

$$\hat{\bar{\psi}}_{a,c,\theta}(\vec{k},\omega) = \hat{\psi}^*_{a,c,\theta}(\vec{k},\omega) \tag{4.64}$$

   L'ondelette est composée de la "version classique" de la CWT (2D+t), adaptée à la translation spatiale et temporelle, plus les trois paramètres d'échelle, de vitesse et de rotation : a, c et θ. L'ondelette spatio-temporelle $\psi_{a,c,\theta}(\vec{x}-\vec{b}, t-\tau) = \psi_{a,c,\theta,\vec{b},\tau}(\vec{x},t)$ est en fait adaptée séquentiellement à l'espace **g** des paramètres considérés de façon séquentielle à pouvoir maximiser l'énergie correspondant à chacun de ces paramètres.
3. On effectue la convolution signal-ondelette dans l'espace de Fourier :

$$\hat{z}(\vec{k},\omega) = \hat{s}(\vec{k},\omega)\; \hat{\bar{\psi}}_{a_0,c_0,\theta_0}(\vec{k},\omega) \tag{4.65}$$

4. On effectue une première FFT inverse 1D $\hat{z}$ par rapport à la variable vecteur d'onde temporel ω.

$$\tilde{z}(\vec{k},\tau) = IFFT_\omega\left\{\hat{z}(\vec{k},\omega)\right\} \tag{4.66}$$

5. On calcule la FFT_{2D}^{-1} de $\tilde{z}(\vec{k},\tau)$ par rapport à la variable vecteur d'onde $\vec{k}$ pour $\tau = \tau_0$:

$$z_{\tau_0}(\vec{b}) = IFFT_{\vec{k}}\left\{\tilde{z}(\vec{k},\tau_0)\right\} \tag{4.67}$$

   $z_{\tau_0}(\vec{b}) = \mathcal{E}^{\mathcal{II}}$ est la densité d'énergie spatiale. Elle permet de remettre à jour l'étage de position spatiale à partir de la nouvelle valeur de la vitesse obtenue à la fin de cette itération t_{i+1}. De la même façon, la densité d'énergie en échelle $\mathcal{E}^{III}$, et donc la nouvelle valeur de l'échelle, est mise à jour après les paramètres de vitesse-rotation et de position, donc au cours de l'itération suivante de la boucle de mise à jour de la vitesse (itération t_{i+2}).

6. On obtient alors la densité d'énergie en vitesse qui permet de remettre à jour l'étage de correction de la vitesse de t_i à t_{i+1}

$$\mathcal{E}^{I}_{a_0,\tau_0}(c_0,\theta_0) = \sum_{\vec{b}\in\mathcal{B}} |z_{\tau_0}(\vec{b})|^2 \tag{4.68}$$

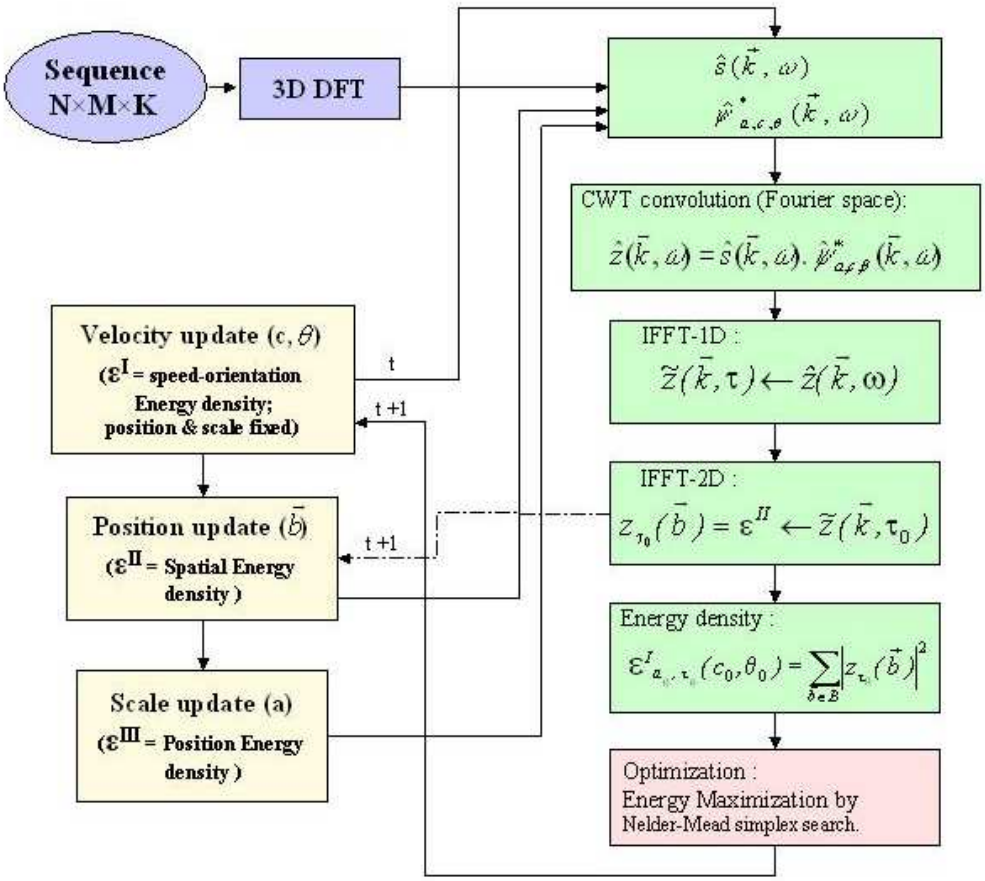


FIG. *4.15 – Algorithme de suivi en vitesse/position/échelle. Une méthode d'optimisation par maximisation des énergies et algorithme du simplexe (Nelder-Mead, voir Annexe I) est utilisée.*

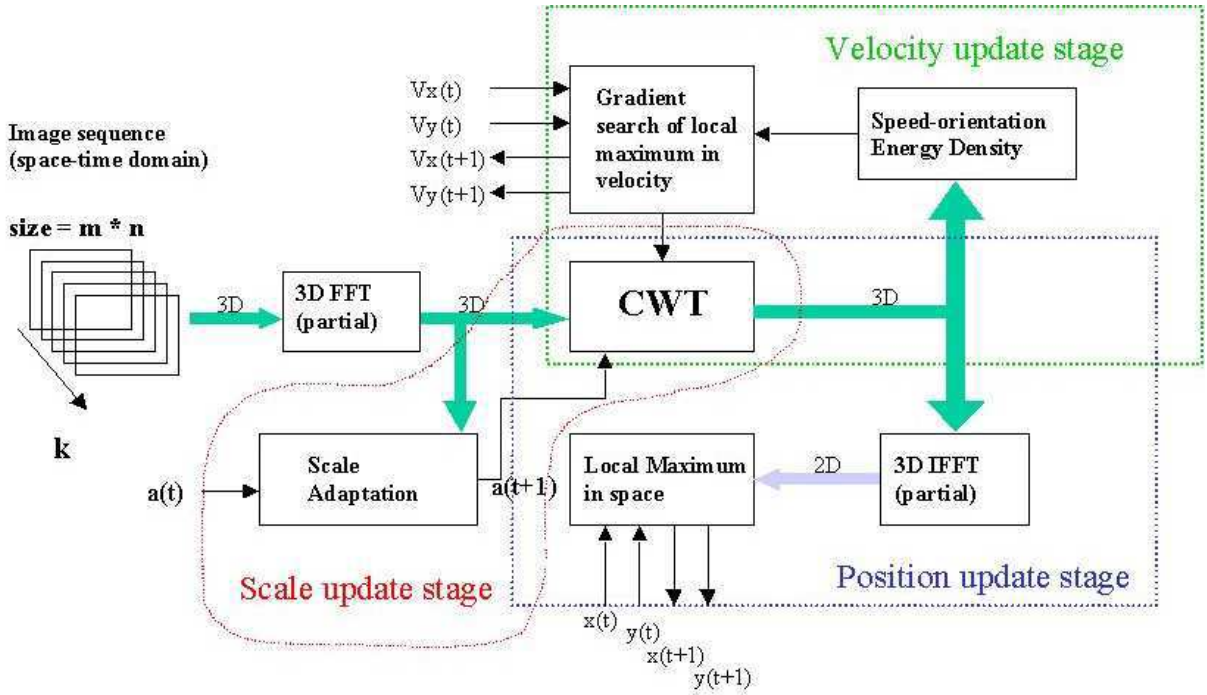


FIG. *4.16 – Autre représentation de l'algorithme de poursuite par CWT adaptée à la vitesse*

Chapitre 5

Résultats et comparaisons

Nous avons utilisé plusieurs types de séquences de test pour notre approche par MTSTWT :
1) Des séquences synthétiques comportant des objets en rotation, translation, changement d'échelle, vitesse constante et accélération constante.
2) Des séquences naturelles : "Tennis player", "Caltrain". La première de ces séquences présente des déplacements locaux simples sur peu de régions de l'image (balle, bras et main du joueur), ainsi qu'un changement d'échelle global (zoom arrière). La deuxième séquence est beaucoup plus complexe et sert de test depuis de nombreuses années, notamment à la mise au point des chaînes de codage-décodage (Ex : la première chaîne numérique complète codage-décodage et transmission satellite MPEG2, mise au point par Philips et présentée pour la première fois à l'IBC, International Broadcast Conference, d'Eindhoven en 1995). Dans cette séquence couleur, aucun objet n'est fixe. Les mouvements sont de type rotation, translation, changement d'échelle global(zoom arrière) et local (rapprochement-éloignement d'objets), de vitesse et d'accélération, dans l'espace 3D et la scène présente de très nombreux détails. En revanche, aucun objet ou région n'est déformé (s'agissant de sa forme initiale 3D). La déformation de perspective ou cisaillement (skew) est peu présente.

5.1 Résultats sur la séquence tennis player

La séquence naturelle tennis player sur laquelle nous avons fait les premiers essais est composée à l'origine de 124 trames couleur (voir base [XIP04] en format SIF (y4m). Nous avons extrait de cette séquence les 30 premières trames. Celles-ci sont réduites, pour nos besoins, à 8 niveaux de gris et les dimensions de chaque trame sont 360×240 (voir figs. 5.1 et 5.2). Les mouvements sont soit locaux (balle et bras du joueur), soit globaux (zoom arrière en fin de séquence). Nous montrons d'abord les résultats d'une analyse qualitative de la scène avec une ondelette adaptée à une seule vitesse. Les mouvements les plus rapides sont extraits avec une ondelette adaptée à une vitesse élevée (proche de celle de la balle à sa vélocité maximale). L'analyse montre que les coefficients ont des valeurs proportionnelles aux vitesses des éléments de la scène. Les vitesses les plus élevées sont rencontrées sur les trajectoires ascendante et descendante de la balle (Figs. 5.4 et 5.5), dans les mouvements de la main ainsi que dans le zoom arrière qui est mis en évidence par la détection du mouvement

de translation du bord de la table (Fig. 5.5). Le sommet du rebond de la balle correspond à une vitesse nulle (ou très faible sur le frame capturé).

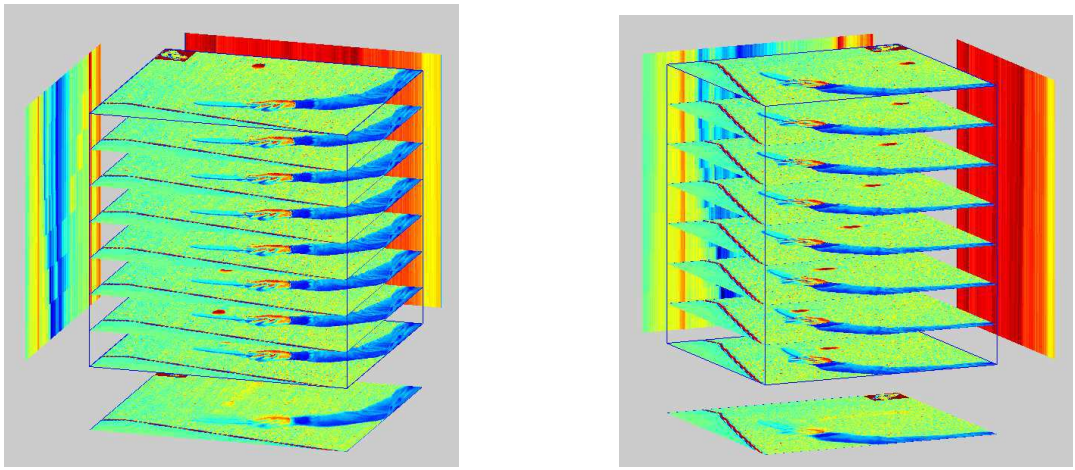


FIG. *5.1 – a) Vue 3D de huit trames successives de la séquence tennis. b) Une autre vue de la séquence tennis montrant la courbe parabolique que suit la balle pendant sa chute et l'accélération gravitationnelle G qui compose cette phase du mouvement.*

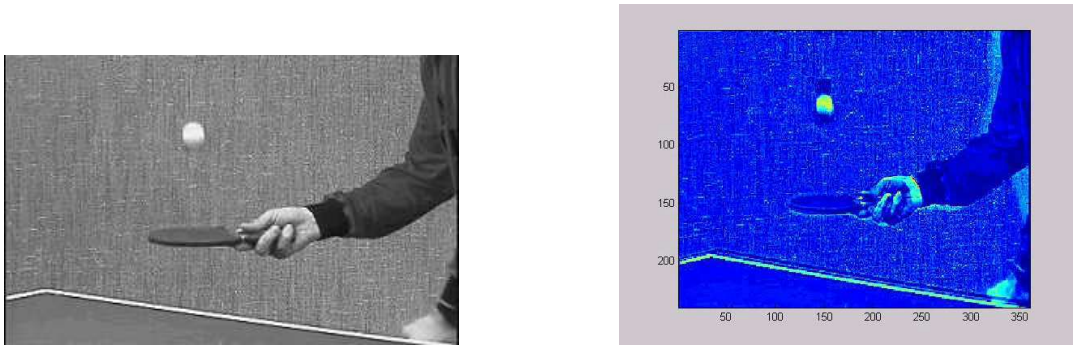


FIG. *5.2 – a) séquence Tennis : trame 1 de la séquence originale b) Détection de vitesse nulle pour l'analyse de la trame 1 : il s'agit de la trame de début de séquence, donc sans information spatiale venant de la trame précédente ce qui rend toute détection de vitesse impossible.*

Dans la première des deux trames qui suivent, la balle est censée être à sa plus grande vitesse. La deuxième trame fait partie de l'action de "zoom arrière" où l'ensemble des objets de la scène diminuent en taille. Le bord blanc, très net, de la table est détecté comme présentant une vitesse $v > 0$. Aussi les éléments de texture du mur de fond de salle sortent du niveau le plus faible (indiqué

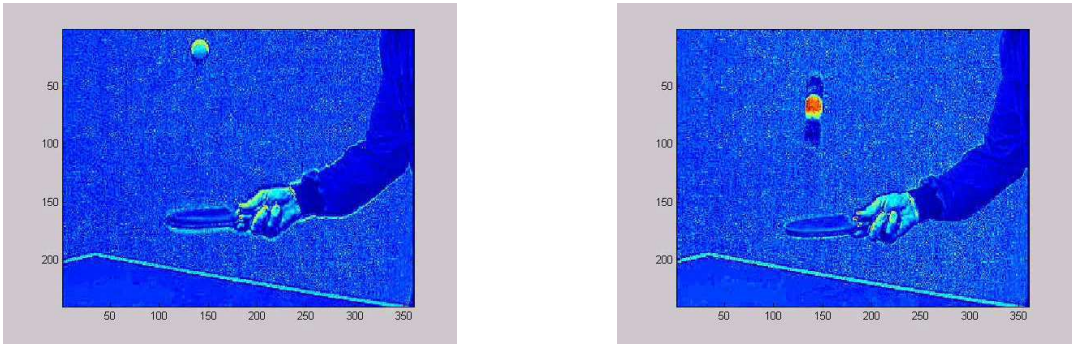


FIG. *5.3 – a) Vitesse nulle au sommet du rebond b) Vitesse élevée pendant la chute .*

par la couleur bleue) pour virer à des tons plus clairs (bleu clair et blanc), ce qui montre que ces éléments de texture sont en déplacement.

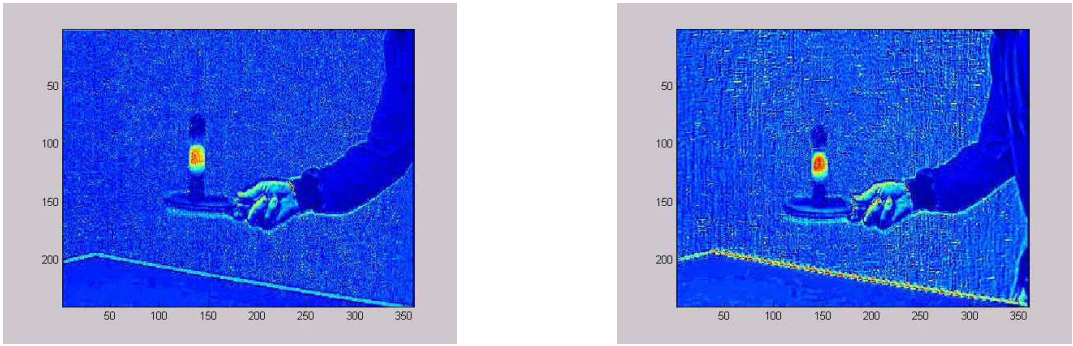


FIG. *5.4 – a) Vitesse maximale après rebond b) Zoom arrière sur la scène : le mouvement du bord de la table est détecté par des coefficients d'amplitude élevée (la bande blanche de bord devient rouge).*

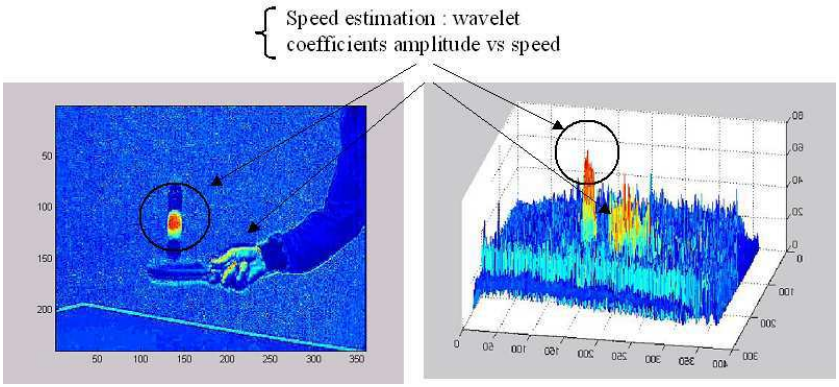


FIG. *5.5 – a) Représentation 2D de la mesure du mouvement b) Représentation 3D des coefficients de l'image analysée en vitesse. Les forts coefficients correspondent aux régions à vitesse plus élevée. On a utilisé la même échelle de couleurs dans les représentations 2D et 3D, ce qui permet de faire la correspondance entre couleur, amplitude des coefficients et vitesse.*

5.2 Analyse de la séquence Caltrain (ou "mobile and calendar")

La séquence Caltrain est plus complexe que les précédentes. Cette séquence a servi de test aux premières chaînes de codage-décodage MPEG2 fonctionnelles. Elle présente des mouvements locaux et globaux. Aucun objet n'est statique, excepté la texture plane de premier plan. Les mouvements locaux sont des translations et rotations en 2D ou 3D réels projetés sur plan 2D (le mobile devant la ligne de train ainsi que les points de la balle représentent ces mouvements 3D projetés). Le mouvement global correspond au suivi du train. Les mouvements de translation et de rotation subissent des accélérations. Il est évident que l'analyse d'une telle scène pour ce qui concerne la quantification et la reconnaissance des mouvements est complexe. En revanche une segmentation par objets est envisageable. Nous montrons néanmoins dans les exemples ci-dessous le comportement en détection de la MTSTWT (voir Figs. 5.6,5.7 et 5.8).

5.3 Complexité de la MTSTWT et temps de calcul

5.3.1 Algorithmes dans les domaines direct et spectral

Nous avons testé l'algorithme de la MTSTWT dans le domaine direct et dans le domaine spectral. La version directe calcule les convolutions de façon séparable dans le domaine direct avec la même longueur de filtre (-4, 4, taille(image)) que dans le domaine spectral. A ce point nous pourrions montrer que calculer la MTSTWT dans le domaine direct est plus rapide que de faire une transformation aller-retour vers le domaine de Fourier en plus du calcul dans ce même domaine. Cependant l'avantage de travailler dans le domaine direct est d'utiliser une représentation "compacte" du filtre en ondelette.

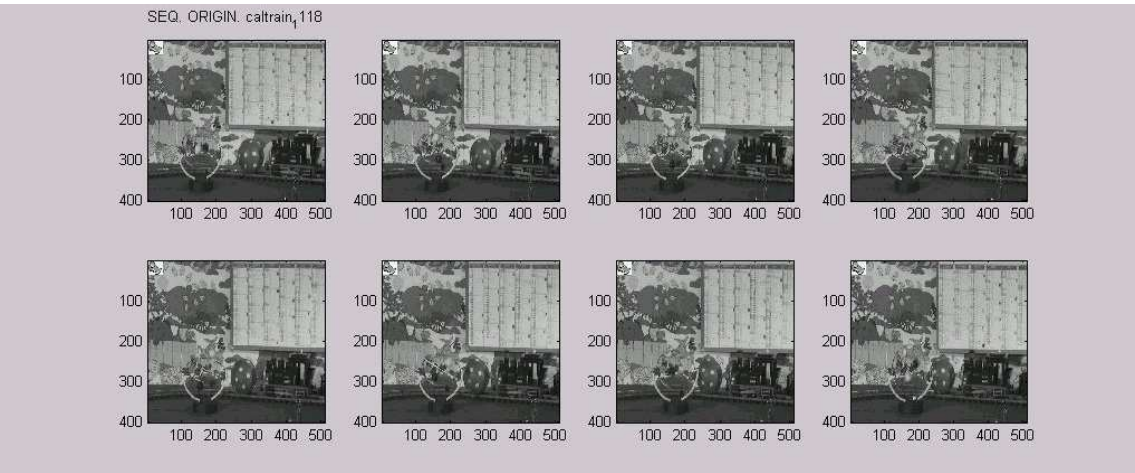


FIG. *5.6 – Nous utilisons un GOF de taille :* $400 \times 512 \times 8$ *, extrait de la séquence caltrain (ou mobile and calendar, provenant de la base de Stephan Wenger, voir [?])). Dans cette séquence complexe, toutes les pixels sont en mouvement. La scène est prise en panoramique de la droite vers la gauche. Tous les mouvements sont présents : translation horizontale de toute la scène, translation du train, rotation et accélération variable de la balle dans deux directions, mouvement complexe du mobile. Nous sommes donc en présence d'une scène dans laquelle l'analyse des mouvements est complexe.*

Un autre point intéressant est que la condition de compacité du filtre, dans ce même domaine, est facilement satisfaite en comparaison de l'algorithme spectral. En effet pour travailler dans le domaine spectral il est préférable de travailler avec une ondelette dont la compacité est bonne à la fois dans le domaine temporel et dans le domaine fréquentiel, ce qui est le cas de l'ondelette de Morlet. Cette transition du domaine direct au domaine spectral est une restriction importante due au principe de Heisenberg.
La version directe, séparable, de l'algorithme de MTSTWT, calculé avec les mêmes longueurs de filtres, est plus lent que la version spectrale. Néanmoins, dans l'espace direct le calcul devrait être plus rapide à condition de pouvoir utiliser des filtres à support plus compact. Dans l'espace direct les filtres peuvent également avoir une longueur inférieure au signal. Dans l'espace de Fourier, ils doivent avoir la même longueur pour une exécution par produits terme à terme.

5.3.2 Résultats

1) Complexité algorithmique de la MTSTWT

• La MTSTWT dans le domaine de Fourier a une complexité de $O(\lambda \times (N^3 LogN))$ où $N = m \times n \times k$ est le nombre de pixels de la séquence et λ est la longueur de la séquence du filtre.

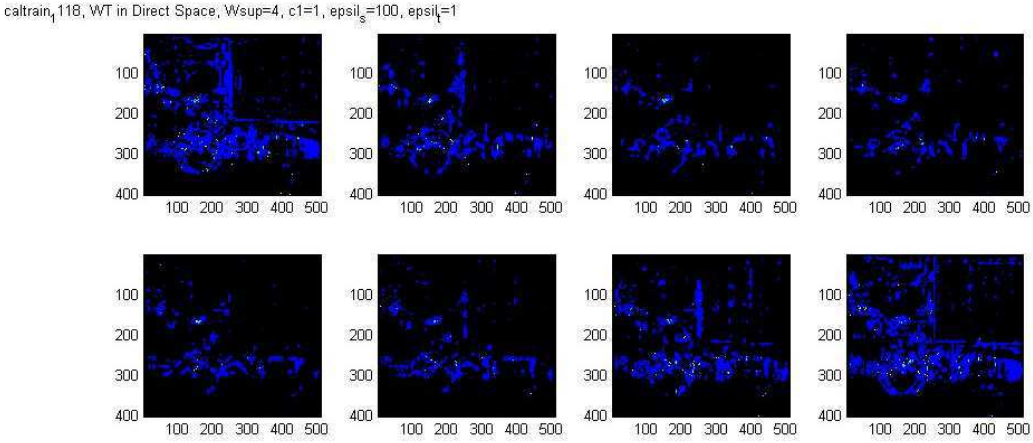


FIG. *5.7 – Analyse par la MTSTWT dans le domaine spectral et les paramètres* $a = 1$, $\epsilon_s = 100$, *sur huit trames de la séquence "Caltrain". Trois points sont intéressants à souligner : 1) la première trame subit un effet de bord dû la convolution non-circulaire (ou la non-symétrisation ou le non-remplissage par une valeur constante) 2) le mouvement accéléré est "détecté" par des coefficients plus forts (plus clairs) lorsqu'on se rapproche de la huitième trame 3) tous les mouvements sont détectés avec des coefficients forts (lumineux) ou faibles selon la valeur de la vitesse. En particulier le logo ("canard" en haut à gauche) du logiciel utilisé pour la décomposition de séquences AVI vers des images compressées JPEG, et qui est statique dans toute la séquence, est totalement invisible.*

• Il faut ajouter une IFFT3 par paramètre de vitesse si l'analyse est réalisée dans le domaine direct, ou une IFFT si l'analyse est réalisée dans le domaine spectral . L'analyse de vitesse est basée sur l'utilisation d'un ensemble d'ondelettes adaptées à différentes vitesses. Le temps d'analyse d'une séquence équivaut à la recherche d'une meilleure base dans un dictionnaire d'ondelettes adaptées à plusieurs vitesses. Le "jeu" de vitesses d'analyse peut être déterminé ou non à l'avance. Un exemple est : $S_c = \{1, 3, 6, 12, 24, 48\}$ pixels/trame.
Ce jeu peut être choisi avec ou sans connaissance a priori sur les vitesses des objets appartenant à la séquence. Une recherche a priori en analyse de scène serait de détecter uniquement les objets possédant une vitesse autour de 3 pixels/trame. Cette notion d'analyse contextuelle peut servir à la compression orientée, ou compression intelligente, dans laquelle les objets seraient plus ou moins compressés selon la gamme de vitesses à laquelle ils appartiennent (cf. thèse d'Isabelle Amonou [Amo04] qui développe ce même concept).
D'autre part les ondelettes, nous l'avons vu, possèdent un paramètre d'anisotropie qui leur donne une faculté d'analyse plus ou moins large autour de la vitesse à laquelle elles sont adaptées. Le terme "autour de" s'entend par le fait qu'un paramètre d'anisotropie est attribué à l'ondelette afin de modifier sa variance et ainsi d'élargir ou de restreindre la gamme de vitesses (ou de fréquences)

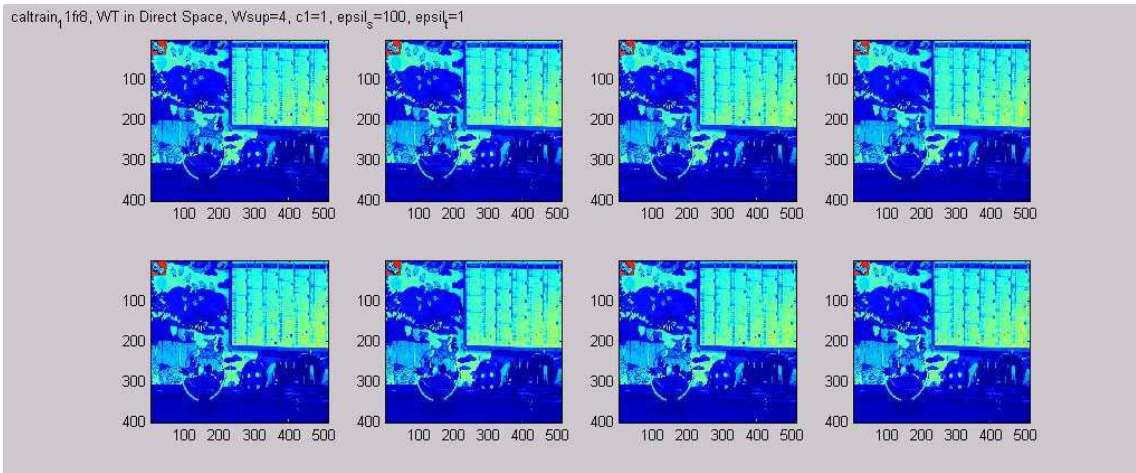


FIG. *5.8 – Analyse par la MTSTWT dans le domaine spectral avec les paramètres* $a = 1$, $\epsilon_s = 100$, *sur un groupe synthétisé à partir de huit trames identiques de la séquence "Caltrain". Ce groupe de trames ne comporte donc aucun mouvement. Nous pouvons facilement voir, en comparaison avec la figure précédente, que la MTSTWT n'a détecté aucun mouvement. Le logo statique, (en haut à gauche), est cette fois visible dans cet exemple par rapport à la figure 5.7.*

qu'elle est en mesure d'analyser. Ce paramètre d'anisotropie donne donc à l'ondelettes sa sélectivité. Ainsi pour une ondelette adaptée à une vitesse de 3 pixels/trame, l'anisotropie ϵ permet de rechercher des vitesses entre 2 et 4 pixels/trame par exemple. La sélection peut finalement être faite par seuillage "dur" ou "mou" (appelées aussi, de façon plus précise, procédures "keep or kill" et "shrink or kill", car on annule dans un cas les coefficients au dessous (ou au-dessus) d'un seuil alors que dans l'autre cas les coefficients au-dessous (respectivement au dessus) du seuil sont augmentés (resp. diminués) de la valeur du seuil).

• Il faut aussi ajouter la FFT3D du bloc de trames analysé préalablement à l'analyse par MTSTWT dans le domaine spectral (qui se fait donc par produit simple de la FFT de la séquence et de la version spectrale de ondelette d'analyse. Cette FFT3D n'est en revanche réalisée qu'une fois pour une gamme de paramètres (plusieurs vitesses et/ou rotations).

2)Vitesse de calcul

• La MTSTWT dans l'espace direct, avec des filtres dont la taille correspond aux dimensions de l'image, c'est-à-dire avec des filtres dont la longueur est la même que dans le domaine de Fourier, offre une vitesse de calcul d'approximativement 30 fois (30 secondes) le temps de calcul dans le domaine spectral (voir ci-dessous les résultats de l'algorithme dans le domaine spectral). Les calculs ont été faits sur une machine Xeon bi-processeur à 2,4GHz (non optimale pour le traitement

d'image).

• La MTSTWT dans le domaine spectral, avec trois bases différentes de vitesse, c'est-à-dire avec trois ondelettes (2D+T) accordées à trois vitesses (3, 6 et 10 pixels/fr), est testée sur un GOF de $360 \times 240 \times 8$ de la séquence Tennis . Le temps de calcul est de 1200ms sur le même Xeon bi-processeur. Il faut ajouter approximativement 380ms pour les IFFT3D, pour chaque vitesse d'analyse, ce qui donne un total de 2400ms, pour une résolution (les valeurs des temps sont données ici pour la résolution la plus élevée qui est aussi la moins rapide en calcul).

• Le calcul rapide du flot optique [Ber99b] entre deux trames de la même séquence et à quatre résolutions prend 10 secondes sur notre machine Xeon.

5.3.3 Conclusion et améliorations sur la MTSTWT

Les ondelettes adaptées au mouvement sont très efficaces pour le calcul de paramètre de mouvement, peuvent être très précises, robustes au bruit et à l'occlusion grâce à leur redondance, et ont la propriété de scalabilité par la construction (multirésolution). Leur application à la compression n'est pas évidente directement pour le calcul de mouvement car :
1) une bonne segmentation des objets est difficile et pas encore bien réalisée particulièrement dans MPEG4.
2) le calcul de la MTSTWT est complexe : son utilisation est semblable à la poursuite d'une meilleure base ("matching pursuit") par le large balayage du champ de paramètres qui doit être fait pour chaque GOF (groupe de frames).

D'autre part :
1) Elles présentent une solution intéressante pour le filtrage de trajectoire post-compensation de mouvement, soit sur un champ dense, soit les trajectoires dans un bloc de trames.
2) Leur efficacité dans l'extraction de paramètre de mouvement peut être employée dans l'analyse de scène.
3) Elles ont déjà montré de bons résultats dans le suivi de cible.

Nos améliorations actuelles sont concentrées sur plusieurs points précis :
1) Utilisation d'ondelettes mieux adaptées à la détection et moins oscillantes (dérivées de gaussiennes, Splines)
2) Calcul de A dans l'espace direct (discuté ci-dessus) afin d'éviter les conversions de l'espace direct à l'espace dual de Fourier (FFT $\longleftrightarrow$ IFFT), coûteuses en temps de calcul.
3) Réduction de la taille du noyau de convolution (particulièrement pour le calcul dans le domaine direct)
4) Utilisation d'un algorithme rapide basé sur "l'algorithme à trous" et/ou calcul dans l'espace direct (convolution dans l'espace euclidien).
5) Extension du cadre cinématique aux filtres ondelettes adaptés à l'accélération ainsi qu'à la déformation d'objets [LCK$^+$98, CLK99]

Chapitre 6

Construction de trajectoires de mouvement

La construction de trajectoires de mouvement dans des scènes animées, en vue d'analyse, de suivi et de compression, a déjà été étudiée notamment par P. Bouthemy et F.G. Meyer, J.P. Leduc, D. Béréziat dans [BHY00, LCK+98, MB94a]. D. Béréziat part d'une séquence d'images, puis calcule, pour des zones d'intérêt sélectionnées au préalable, une famille de courbes trajectoires par intégration de champs de vecteurs vitesse estimés sur la séquence. Dans [SRT93], le calcul des trajectoires à partir de vecteurs d'intérêt du flot est réalisé. Les trajectoires sont isolées en ensembles (sets) appartenant à différents objets. Ceci est réalisé avant que soit effectuée toute interprétation de forme ou de mouvement. Si l'on considère indépendantes les trajectoires des objets, le point de concours (FOE, Focus Of Expansion) de toutes les trajectoires appartenant à un même objet, détermine et caractérise de façon unique cet objet. Ce FOE permet ainsi de segmenter les trajectoires caractérisant des objets individuels.
La détection de ruptures dans des signaux et des séquences, basée sur une identification rapide de polynôme, a été proposée récemment aussi par M. Fliess et al. [FSR03] .

6.1 Construction d'une trajectoire par objet ou par bloc

Nous proposons ici un schéma d'identification de trajectoire et de prédiction de mouvement basé tout d'abord sur l'acquisition des paramètres de mouvement avec les filtres spatio-temporels présentés précédemment. L'idée est d'identifier la trajectoire d'un objet, ou de plusieurs, en se basant sur les observations et paramètres acquis au cours du GOP passé. Cette trajectoire doit pouvoir être identifiée, dans la plupart des cas, à un polynôme d'ordre, ou à une Spline, d'ordre (n). La deuxième phase consiste à utiliser cette trajectoire construite sur le GOP pour prédire sur la ou les trames qui suivent, la position de l'objet. L'approche se distingue ici nettement des techniques BM pour lesquelles l'analyse du mouvement reste extrêmement sommaire et n'est basée que sur une seule translation de bloc. De plus ce mouvement ne coïncide pas nécessairement avec le mouvement d'un objet, est indépendante du mouvement des blocs voisins ce qui présente une certaine incohérence du point de vue de la modélisation du mouvement de l'objet par les vecteurs, et enfin attribue à

tous les pixels du bloc le même vecteur, indépendamment de la forme de l'objet.

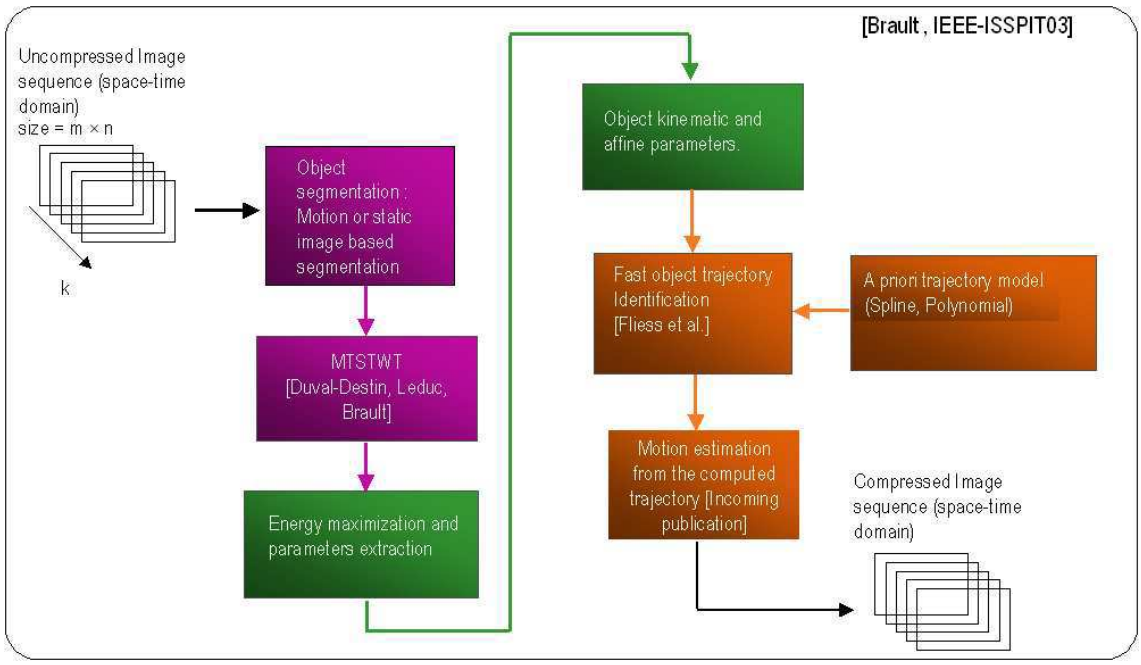


FIG. *6.1 – Proposition d'un schéma d'estimation/prédiction de mouvement contextuelle orienté objet. Ce schéma est basé dans un premier temps sur la détection des paramètres cinématiques des objets par une transformée de type MTSTWT. Dans un deuxième temps on cherche à identifier la trajectoire d'un (ou plusieurs) objet à un polynôme, ou une spline, dont les paramètres sont déduits des paramètres acquis par la MTSTWT. Finalement la prédiction de la position de l'objet dans la ou les trames suivantes peut alors se faire sur la base de la trajectoire identifée dans le GOP passé. Cette approche de prédiction/compensation présente donc la double caractéristique d'être une approche véritablement orientée objet et pouvant offrir une prédiction de mouvement beaucoup plus évoluée et performante que les approches de type BM.*

Conclusion à la première partie

Le but de cette partie concernant l'estimation de mouvement a été de présenter une approche contextuelle, c'est-à-dire prenant en compte les caractéristiques d'une scène, avant d'en réaliser la compression. Il nous a notamment paru intéressant, dans le cadre de la norme MPEG4 V2 qui existait lorsque cette thèse a été débutée, de considérer différemment l'aspect orienté-objet de cette norme. Dans MPEG4 V2, les motifs d'intérêt sont segmentés en "objets", mais leur codage spatial est toujours réalisé par DCT et l'estimation de mouvement est toujours réalisée, sur les mouvements des blocs, par de simples vecteurs de translation entre deux trames de la séquence.
L' approche par blocs présente l'avantage d'une certaine simplicité de mise en oeuvre et d'une bonne robustesse. Néanmoins, elle reste très limitative quant aux développements futurs possibles dans une vision avancée de la compression. De même que les méthodes de compression vidéo ont été révolutionnaires lorsqu'elles ont intégré la réduction de redondance temporelle par mouvements de blocs, de même nous pensons que ces techniques méritent aujourd'hui le passage de l'aspect systématique, comme pour la norme H264, à l'aspect contextuel orienté compréhension de la scène, pour devenir vraiment souples et performantes.
Le prix à payer pour un développement de la compression dans ce sens est élevé et la première norme MPEG4 intégrant les premiers aspects contextuels ainsi que un codage par réalité virtuelle (synthèse de l'image), n'a pas eu le succès escompté.
En ce qui concerne le travail réalisé dans cette partie, il a consisté à utiliser une des méthodes qui pouvait permettre une réduction de la redondance temporelle par une estimation "intelligente" de la scène. En particulier, nous nous sommes intéressés à des méthodes permettant d'obtenir les paramètres de mouvement pour différents motifs (objets ou régions) de la scène.
Plusieurs méthodes ont déjà été abordées pour obtenir un champ dense du mouvement ou une cartographie des mouvements par objets. Parmi elles, le flot optique et le filtrage spatio-temporel. Nous avons choisi d'investiguer la deuxième approche en utilisant des familles d'ondelettes peu connues jusqu'à aujourd'hui et qui présentent l'intérêt de pouvoir détecter et quantifier plusieurs types de mouvement ainsi que de fournir les paramètres cinématiques des régions en mouvement dans une scène. Nous avons, pour cela, codé et mis en oeuvre une transformée en ondelettes spatio-temporelles adaptée au mouvement, et en particulier à la vitesse, dans l'espace réciproque de Fourier. Nous avons proposé l'adaptation de cette transformée au domaine direct et l'utilisation d'ondelettes adaptées dans un schéma de calcul rapide de la T.O. redondante, "l'algorithme à trous", afin de diminuer les temps de calcul. Nous avons comparé les temps de traitement de ces filtres spatio-temporels avec la méthode rapide de calcul du flot optique développée dans la thèse de C. Bernard [Ber99b].

Enfin, nous avons proposé de réaliser une estimation de mouvement basée sur une *estimation de trajectoire*. En effet, l'analyse par filtrage spatio-temporel s'effectue sur quelques trames d'une séquence et permet d'acquérir les paramètres cinématiques d'objets d'intérêt. Une fois ces paramètres correctement acquis, nous proposons d'identifier la trajectoire que suivent les objets d'intérêt, à un modèle cinématique. Ce modèle serait alors paramétré grâce au résultat de l'estimation et la quantification du mouvement réalisé par les filtres ondelettes adaptés au mouvement que nous avons présentés dans cette partie. Ceci permettrait de prendre en compte des mouvements complexes analysables par ces mêmes transformations, à savoir : mouvement uniforme, mouvement uniformément accéléré, rotation, translation, changement d'échelle (rapprochement-éloignement ou déformation), et tout autre forme plus complexe de mouvement.

Partie II : Segmentation et estimation de mouvement par modèles de Markov cachés et approche bayésienne dans les domaines direct et ondelettes

Introduction à la seconde partie

Dans la deuxième partie de ce mémoire nous nous sommes intéressés à la segmentation non-supervisée d'images et de séquences d'images par des méthodes bayésiennes mises en oeuvre dans le domaine direct ainsi que dans le domaine transformé des coefficients d'ondelettes. Dans le domaine du traitement d'image, les méthodes de segmentation sont aujourd'hui nombreuses. Afin de pouvoir situer notre travail parmi ces méthodes, nous les avons d'abord classées en deux grandes familles : les approches basées contour et les approches basées région [CP95, SHB99].

1) Approches basées contour :
L'extraction des contours fournit un premier moyen d'obtenir une segmentation d'image. La segmentation par le contour peut se décomposer elle-même en quatre approches principales. Ce sont :
a) l'approche surface et les méthodes basées sur la morphologie [Har84, Can86, Der87]
b) les approches par contours fermés
c) la modélisation markovienne et les algorithmes déterministes d'optimisation de maximum *a posteriori* comme la non-convexité graduelle (GNC) [SCZ89] et le recuit de champ moyen(MFA) [ZC93]
d) les méthodes variationnelles comme les contours actifs (snakes), utilisés récemment dans le suivi de scènes dynamiques [JDBA00].

2) Approches basées région :
Dans cette deuxième famille, nous pouvons distinguer aussi quatre groupes basés sur la classification.
- Le premier groupe correspond aux méthodes n'utilisant qu'un seul attribut de l'image (en général le niveau de gris) et sont qualifiées de mono-dimensionnelles. Ces méthodes reposent sur l'exploitation de l'histogramme à partir duquel on cherche à trouver des seuils qui définiront les différentes classes. La recherche de minima et l'approximation des modes par des gaussiennes permet de définir ces seuils qui peuvent être classés en trois catégories : le seuillage global (méthode de Fisher, méthode de Bhattacharya, voir [CP95] pp. 242 et 245), le seuillage local (méthode bayésienne de Mardia et Hainsworth, voir [CP95] p.247) et le seuillage inter-modes . Lorsque le seuil est unique, l'opération de seuillage est appelée "binarisation". Lorsque les seuils sont multiples, on parle alors de multiseuillage ou de "classification".
- Le second groupe est basé sur des méthodes multi-dimensionnelles qui consistent à classifier les pixels à partir non plus d'un seul attribut mais à partir d'un ensemble d'attributs. Les attributs d'une image sont nombreux. Ce sont, entre autres, la texture (Rao [Rao93], Julesz [Jul83, JB83],

Haralick [HSD73], Gagalowicz [GG00]), les attributs stochastiques de l'image (moyenne, moments, autocorrélation), les attributs d'une région (moments et moyennes) (Gagalowicz 83), les matrices de cooccurrence (Haralick [AH98], Conners et Harlow [CH80]), la décomposition dans un espace transformé (Fourier, ondelettes), le contraste (Zeboudj [JPZ88]), les attributs fractals (Pentland [Pen84], Voss [Vos86]) et multifractals (Arnéodo [AAE+95], Voss [Vos86], Keller[KC89]), les attributs surfaciques comme la courbure ou la surface polynômiale du second ordre (quadrique) (Peet et Sahota [PS85]). La classification multidimensionnelle se sépare en deux groupes : les méthodes avec apprentissage et les méthodes non-supervisées (Gagalowicz [GG00], Coleman et Andrews [CA79]) .

- Le troisième groupe représente les méthodes par regroupement (clustering) [Boc04], hiérarchiques ou non, comme les algorithmes K-means, Fuzzy C-means, ou "CLUSTER" qui sont aussi appelées communément "apprentissage non-supervisé" ([JD88] pp. xiii et 134) et fonctionnent par organisation des données selon une structure fondamentale qui regroupe les individus ou crée une hiérarchie de groupe.

- Le quatrième groupe est celui qui concerne notre approche. Il est basé sur la modélisation statistique et en particulier sur la modélisation markovienne et sur l'estimation, supervisée ou non. Parmi les travaux importants en segmentation supervisée d'images texturées, nous devons citer celui de Geman et Geman [GG84, GGG87] qui utilisent la régularisation et le pseudo maximum de vraisemblance (pML). Plus récemment, la modélisation markovienne a été réalisée dans le domaine transformé des ondelettes et s'est concrétisée par le développement de nombreux modèles qui exploitent les propriétés particulières de la distribution des coefficients d'ondelettes. Avant de présenter plus avant les méthodes statistiques basées région, et en particulier les méthodes markoviennes et l'utilisation du domaine transformé des ondelettes, nous pouvons ajouter à ces deux grandes familles d'approches de la segmentation les méthodes de division hiérarchiques. Ce sont notamment le quadtree, utilisé par exemple pour la compression vidéo [KLM04] et les méthodes structurales de partitionnement élémentaire (régulier ou non) comme les diagrammes de Voronoï. Ces méthodes de division hiérarchique servent plutôt de base à la segmentation.

Une brève présentation des méthodes de segmentation statistique pourrait commencer par les champs aléatoires gaussiens multivariés de Markov (GMRF), modèles qui ont été intensivement appliqués pour la segmentation d'images fixes [Haz00]. Depuis plusieurs années, les approches bayésiennes multiéchelle ont montré de bons résultats dans le domaine de la segmentation d'images fixes ainsi que, récemment, dans le domaine de la segmentation de séquences animées. Supervisée, partiellement-supervisée et non-supervisée, la segmentation d'images fixes avec des approches bayésiennes multi-échelles a été largement étudiée et décrite dans la littérature. Dans le cadre bayésien, les approches multi-échelles ont montré qu'elles peuvent efficacement intégrer des attributs d'image aussi bien qu'une information contextuelle pour la classification. Les caractéristiques des images peuvent être représentées par différents modèles statistiques et l'information contextuelle peut être obtenue en employant les modèles multi-échelle comme, par exemple, la dépendance des étiquettes de pixels entre les échelles dans une approche multi-échelle [BS94]. D'autre part, des ap-

proches multi-contextuelles ont été développées conjointement avec une segmentation multi-échelle (JMCMS Joint MultiContext and Multi-Scale). Elles emploient la fusion des informations intra et inter-échelles [FX01]. Pour fusionner l'information contextuelle multi-échelle, plusieurs méthodes ont été employées dont l'utilisation d'un champ aléatoire multiéchelle (MRF) combinée avec un estimateur séquentiel de type maximum a posteriori (SMAP) [BS94, CB01a].

Plus récemment, le transfert du modèle observé à un domaine dual comme le domaine des ondelettes a permis de tirer profit de la propriété très intéressante des coefficients d'ondelettes à être modélisés : intra-échelle, par un mélange de gaussiennes indépendantes (IGM) et inter-échelles, en considérant l'évolution et les capacités "d'essaimage" des coefficients et de leur voisinage à travers les échelles. Des modèles utilisant l'évolution des coefficients entre les échelles par arbre de Markov caché (HMT) [CNB98, RCB01], ainsi qu'une version améliorée utilisant les dépendances inter sous-bandes, à une échelle donnée (modèle Hmt-3s) ont été développés [FX02]. Dans [CB01b] une segmentation multiéchelle a été développée sur la base d'un HMT et de la fusion inter-échelle de l'information contextuelle. Récemment une autre approche pour la segmentation non-supervisée et utilisant un modèle de Markov caché (HMM, Hidden Markov Model) dans le domaine des ondelettes a été utilisée conjointement avec une méthode de regroupement [SF03b, SF03a].

Le Groupe des problèmes inverses (GPI/LSS) a développé récemment une méthode de segmentation non-supervisée basée sur une modélisation Markovienne dans le domaine direct de l'image [FMD03, FMD05]. Cette méthode emploie un HMM pour les étiquettes des classes assignées aux différentes régions d'une image. La différence entre le modèle basé HMT et le modèle HMM est que le premier modélise les caractéristiques de l'image, en l'occurrence les coefficients du domaine des ondelettes, à la fois intra et inter échelles. Le modèle HMM, en revanche, est basé sur les caractéristiques "contextuelles" de l'image que sont les étiquettes attribuées aux pixels, dans le domaine direct de cette image. Nous verrons aussi que l'approche du GPI, sur laquelle nous avons basé notre travail, utilise un modèle de Potts (PMRF, Potts-Markov random field) afin de rendre plus ou moins homogènes les régions segmentées. Ce modèle, qui nous a servi de base, est souvent référencé par la suite comme modèle BPMS (Bayesian Potts-Markov Segmentation).

Sur la base de ce modèle BPMS, nous avons réalisé une première application de segmentation d'une séquence vidéo $2D + t$ basée sur la segmentation "incrémentale" des images de la séquence [5]. Cette première application, décrite au chapitre 8, a permis de montrer que l'on pouvait accélérer de façon significative la segmentation d'images lorsque la différence entre celles-ci est faible, ce qui est souvent le cas des images d'une même scène vidéo. Nous montrons dans ce même chapitre une application directe à l'estimation de mouvement ainsi qu'à la compression de séquences.

Cependant, bien que la méthode de segmentation "directe" (BPMS) permette d'obtenir une bonne qualité de segmentation, nous avons recherché le moyen d'augmenter la vitesse de segmentation, à la fois pour des images fixes et pour des séquences d'images. Ceci a motivé une nouvelle approche basée sur la projection de l'image dans le domaine transformé des ondelettes pour y réaliser la segmentation. En raison de certaines propriétés particulières dont nous parlerons en détail dans le

chapitre 9, les coefficients d'ondelette, obtenus par décomposition sur une base orthogonale, peuvent être modélisés par un modèle de mélange de gaussiennes (IGMM) [CNB98, RCB01, PKLH96]. Une première distribution y représente les forts coefficients, d'importance majeure mais en faible nombre. Une seconde distribution représente les coefficients de faible importance et amplitude mais en grand nombre. Ainsi la segmentation de ces coefficients d'ondelette peut être réalisée en utilisant seulement 2 classes distinctes. D'autre part, les coefficients faibles sont éliminés par seuillage, fort ou faible, dans des approches de "débruitage". Nous verrons que cette méthode va s'appliquer de façon plus "radicale" dans notre cas, puisque nous ne nous intéressons qu'à la "détection" de régions comportant peu d'informations de détail, qu'est la segmentation. Notre approche est, en ce sens, différente des approches de débruitage utilisant des mélanges de gaussiennes par échelle (SMG, Scale Mixture of Gaussians), comme par exemple dans [PSWS03]. La prise en compte des propriétés spécifiques des coefficients d'ondelettes, dans notre approche de Potts-Markov non-supervisée et avec les modèles et solutions adoptés, va nous conduire finalement à une réduction importante des temps de segmentation. Afin de rendre encore plus efficace la segmentation dans le domaine des ondelettes, nous avons aussi modifié le modèle du PMRF afin de l'adapter aux orientations privilégiées des sous-bandes d'ondelettes. Ce nouveau modèle utilise alors, non plus seulement un voisinage d'ordre 1, mais aussi un voisinage d'ordre 2 correspondant aux directions diagonales des sous-bandes d'ondelettes.

Cette seconde partie est organisée de la façon suivante. Nous présentons dans le chapitre 7 un modèle de segmentation bayésienne non-supervisée développé récemment au GPI. Ce modèle utilise un champ aléatoire de Potts-Markov pour les étiquettes des pixels ainsi qu'un algorithme itératif de type MCMC (Markov Chain Monte Carlo) avec échantillonneur de Gibbs. Il est mis en oeuvre dans le domaine direct de l'image (sans projection dans un espace dual). Le chapitre 8 décrit un développement simple de cette approche de façon à l'adapter à la segmentation de séquences d'images. Cette segmentation utilise le modèle décrit dans le chapitre 7 et est réalisée de façon "incrémentale". Ceci permet un gain de temps considérable dans la segmentation des images d'une même scène dont les variations sont en général relativement faibles. Nous montrons également comment la segmentation peut servir à la compression d'images et de séquences, ainsi qu'à l'estimation de mouvement. Le chapitre 9 reprend le modèle initial dans le domaine direct pour le développer dans le domaine des ondelettes. L'intérêt de la projection dans un domaine transformé réside dans l'apport de propriétés particulières au domaine choisi. Notre but étant de réduire le coût de calcul de l'approche dans le domaine direct, nous avons trouvé dans le domaine des coefficients d'ondelettes les propriétés qui permettent de simplifier l'algorithme initial et de diminuer de façon très significative la vitesse de segmentation. Afin d'adapter le modèle PMRF aux sous-bandes d'ondelettes orthogonales nous présentons un développement du modèle de Potts qui suit les directions privilégiées de la décomposition orthogonale. En plus du voisinage d'ordre 1, notre nouveau modèle de Potts prend en compte un voisinage d'ordre 2 qui correspond mieux aux-sous-bandes diagonales d'ondelettes. Enfin le chapitre 10 présente les résultats comparatifs de notre nouvelle méthode de segmentation bayésienne dans le domaine des ondelettes par rapport à la méthode dans le domaine direct et aux méthodes de type HMT et K-means.

Chapitre 7

Modélisation markovienne pour la segmentation bayésienne

L'approche de segmentation et fusion conjointe par méthode probabiliste bayésienne, et utilisant un modèle de Markov caché (Hidden Markov Model, HMM) et champ de Potts, a été développée au sein du Groupe des problèmes inverses (GPI/LSS Supélec) et bien décrite dans [FMD03, FMD05]. Nous présentons dans ce chapitre les grandes lignes de l'approche bayésienne pour la segmentation dans une hypothèse d'indépendance intra-classe des pixels. En effet, le modèle décrit dans [FMD05], et sur lequel nous avons basé les travaux décrits dans ce manuscrit, est sans cesse en évolution et depuis ces travaux qui nous ont servi de base, a évolué en un modèle où les pixels intra-régions présentent une dépendance locale et sont modélisés par un champ de Markov d'ordre un. Cette modélisation a été utilisée dans différentes travaux récents au GPI, notamment en segmentation combinée fusion (O. Féron [FMD03, FMD05]), en segmentation hyper-spectrale (A. Mohammad-pour [MFMD04]), en super-résolution (F. Humblot [HMD05]) et en séparation de sources (M. Ichir [IMD04]). L'évolution des modèles, au sein du GPI, continue et réside essentiellement dans le raffinement des hypothèses et dans la prise en compte de modèles de plus en plus complexes.

La base de l'approche bayésienne avec modèle de Potts-Markov repose principalement sur le choix des hypothèses (lois a priori), locales (régions et bruit local) ou globales (image complète et bruit global), sur les modèles de lois de luminance "homogène" dans les régions et sur le modèle de bruit, local et global. Nous présentons et justifions l'intérêt de la modélisation markovienne et de l'utilisation d'un champ de Potts. Nous détaillons les principales étapes du calcul des lois a posteriori et leur estimation par méthode de Monte Carlo et échantillonnage de Gibbs.

Le problème de segmentation peut être classé dans les problèmes inverses. Connaissant une image observée et bruitée g nous cherchons à déterminer une segmentation en régions homogènes de l'image d'origine, non bruitée, f. Les approches possibles pour la résolution des problèmes inverses sont nombreuses. Celles ci commencent par l'inversion simple puis l'inversion généralisée et les méthodes de régularisation déterministes. Une première voie, déterministe, se détache des approches possibles. Elle peut se décomposer globalement en quatre méthodes : analytique, par développement

en séries, par décomposition sur une base et algébrique. Cependant toutes ces méthodes ne sont pas toujours satisfaisantes et une deuxième voie, probabiliste, va présenter simultanément les avantages de :
1) prendre en compte les erreurs dues à la discrétisation, aux mesures et à la modélisation.
2) prendre en compte les informations a priori sur l'observable, sur l'image, sur le bruit.
3) caractériser l'incertitude qui reste dans la solution proposée.
La voie probabiliste compte aussi plusieurs "classes" de méthodes, parmi lesquelles l'estimation statistique bayésienne. Celle-ci est le support de nombreux développements et modélisations réalisés au GPI et est à la base de notre développement de la segmentation. Les quelques lignes d'introduction à la segmentation sont largement détaillées et nous citons ici quelques excellentes références pour une bonne approche des problèmes inverses et de l'estimation bayésienne [Dem02, MD01, MD04, Idi01, Jay95, Mac00, Rob92, Rob96].

7.1 Brève introduction à l'approche de la segmentation bayésienne

L'approche initiale de la segmentation bayésienne sur laquelle nous avons basés les travaux décrits plus loin peut être ainsi décrite. En adoptant des hypothèses sur le bruit et sur l'image, ainsi qu'un modèle de Markov caché pour les segments, puis en appliquant la règle de Bayes, nous obtenons une expression de la loi a posteriori sur l'image. Dans le cas d'un modèle linéaire gaussien, l'expression de cette loi a posteriori est aussi gaussienne, loi qui est définie par ses deux premiers moments. L'expression est alors calculable analytiquement. Lorsque le modèle n'est pas gaussien, nous sommes dans le cas général où les solutions sont :
1) soit d'approximer la loi a posteriori par une gaussienne dont on calcule la moyenne et la matrice de covariance
2) soit de définir un estimateur ponctuel à partir de cette loi. Cet estimateur peut être le maximum a posteriori (MAP), la moyenne a posteriori (PM ou posterior mean) ou le maximum a posteriori marginal (MAP marginal). Les estimateurs PM et MAP marginal nécessitent le calcul d'intégrales de dimension élevée, donc difficile analytiquement. En revanche, le MAP peut se calculer par des techniques d'optimisation [MD04]. Afin de pouvoir calculer un estimateur de type PM, une solution est d'échantillonner la loi a posteriori. Pour cela nous utilisons le principe des techniques de Monte Carlo [0] (voir une introduction dans [Dem02], Annexe D.3).
Ces techniques permettent (MacKay chap. 23 [Mac00]) de résoudre deux problèmes :
1) d'engendrer des échantillons $\{\boldsymbol{f}^{(r)}\}_{r=1}^{R}$ à partir d'une distribution $P(\boldsymbol{f})$
2) d'utiliser ces échantillons pour calculer des estimations. En effet, d'une manière générale, l'espérance d'une fonction $\phi(\boldsymbol{f})$ suivant la loi $p(\boldsymbol{f}|\boldsymbol{g})$:

$$E(\phi(\boldsymbol{f})) = \int \phi(\boldsymbol{f}) p(\boldsymbol{f}|\boldsymbol{g}) df$$

[0] Le nom des méthodes de Monte Carlo provient du célèbre casino (nom italien) et de l'utilisation des nombres aléatoires. Il fut probablement utilisé, à l'origine, en 1947 par Nicholas Metropolis au cours du projet de la seconde guerre mondiale "Manhattan", projet de simulations sur la fission nucléaire, à Los Alamos.

Une des méthodes de Monte Carlo est l'échantillonneur de Gibbs. Prenons l'exemple d'une loi à deux variables $\boldsymbol{f} = (f_1, f_2)$. Pour chaque itération, on démarre de l'état présent $\boldsymbol{f}^{(t)}$ et f_1 est échantillonné à partir de la loi conditionnelle $P(f_1|f_2)$, avec $f_2 = f_2(t)$. Dans un deuxième temps, c'est la valeur de f_2 qui est déterminée à partir de $P(f_2|f_1)$, avec $f_1 = f_1(t)$, en utilisant la valeur de f_1 que l'on vient de calculer. Le nouvel état $\boldsymbol{f}^{(t+1)}$ est obtenu par les deux nouvelles valeurs de f_1 et f_2 et complète l'itération de Gibbs. Dans un système à K variables, une iteration complète demande d'échantillonner un paramètre à la fois :

$$\left\{\begin{array}{lcl} f_1^{(t+1)} & \sim & P(f_1|f_2^{(t)}, f_3^{(t)}, \ldots, f_K^{(t)}) \\ f_2^{(t+1)} & \sim & P(f_2|f_1^{(t)}, f_3^{(t)}, \ldots, f_K^{(t)}) \\ \quad\vdots & & \\ f_K^{(t+1)} & \sim & P(f_K|f_1^{(t)}, f_2^{(t)}, \ldots, f_{K-1}^{(t)}) \end{array}\right. \tag{7.1}$$

Les grandes lignes de l'approche bayésienne peuvent être résumées par les six points suivants :
• On écrit la relation de base entre l'image observée $g(\boldsymbol{r})$, l'image recherchée $f(\boldsymbol{r})$ et le bruit $\epsilon(\boldsymbol{r})$:

$$g(r) = f(r) + \epsilon(r), \quad r = (x, y) \qquad , \qquad \boldsymbol{g} = \boldsymbol{f} + \boldsymbol{\epsilon}$$

• On émet une hypothèse le bruit sur $\boldsymbol{\epsilon}$ ce qui nous permet d'exprimer $p(\boldsymbol{g}|\boldsymbol{f}) = p_\epsilon(\boldsymbol{g} - fb)$
• On émet une hypothèse sur l'image d'origine $\boldsymbol{f}$ puis on la traduit par la loi a priori $p(\boldsymbol{f})$
• On applique la règle de Bayes qui permet de calculer la loi a posteriori : $p(\boldsymbol{f}|\boldsymbol{g}) = \frac{p(gb|fb)p(fb)}{p(gb)}$
• On utilise cette loi a posteriori pour définir une solution pour le problème, par exemple l'estimation au sens du MAP (Maximum a posteriori) :

$$\hat{\boldsymbol{f}} = \arg\max_{\boldsymbol{f}} p(\boldsymbol{f}|\boldsymbol{g}) = \arg\min_{\boldsymbol{f}} J(\boldsymbol{f}) = -\ln p(\boldsymbol{f}|\boldsymbol{g})$$

• Dans le cas du MAP, l'optimisation peut se faire de différentes manières en fonction de la nature de $J(\boldsymbol{f})$. Si $J(\boldsymbol{f})$ est quadratique (cas gaussien), on peut obtenir une solution analytique. Si $J(\boldsymbol{f})$ est convexe, on peut utiliser n'importe quel algorithme de descente (gradient, gradients conjugués). En revanche si $J(\boldsymbol{f})$ n'est pas convexe, il faut faire appel à des techniques d'optimisation plus sophistiquées : GNC (Graduated Non-Convexity), recuit simulé, etc.).
Si, par contre, on choisit d'autres estimateurs, par exemple la moyenne a posteriori (PM), l'outil de base est les méthodes MCMC.

7.2 Segmentation bayésienne dans le domaine direct

La segmentation d'une image bruitée commence par un premier objectif qui est de trouver une forme non bruitée $f(\boldsymbol{r})$ de l'image observée $g(\boldsymbol{r})$. L'image observée, bruitée, répond au schéma ci-dessous (fig. 7.1) :

FIG. *7.1 – a) Image observée g b) image originale f c) bruit ϵ*

et l'observable $g(\boldsymbol{r})$ est donc donné par la relation :

$$g(\boldsymbol{r}) = f(\boldsymbol{r}) + \epsilon(\boldsymbol{r}), \qquad \boldsymbol{r} \in \mathcal{R} \tag{7.2}$$

où $\mathcal{R}$ est l'ensemble des sites $\boldsymbol{r}$ de l'image.
L'approche bayésienne consiste alors à formuler un certain nombre d'hypothèses $\mathcal{H}$ concernant le problème.
Prenons comme **première hypothèse** que le bruit $\epsilon(r)$ est gaussien, centré et blanc. Cette hypothèse reflète le fait que la seule information a priori que nous pouvons avoir est que son énergie (variance $v_{epsilon}$) est fixée. Cette information nous permet alors d'écrire :

$$p(\epsilon(\boldsymbol{r})) = \mathcal{N}(\epsilon(\boldsymbol{r})|0, v_\epsilon) \qquad \Longrightarrow \qquad p(g(\boldsymbol{r})|f(\boldsymbol{r})) = \mathcal{N}(g(\boldsymbol{r})|f(\boldsymbol{r}), v_\epsilon) \qquad \forall \boldsymbol{r} \in \mathcal{R}$$

où $v_\epsilon = \sigma_\epsilon^2$ est la variance de bruit.
En adoptant la notation vectorielle simplificatrice suivante : $\mathbf{g} = \{g(\mathbf{r}), \mathbf{r} \in \mathcal{R}\}$, $\mathbf{f} = \{f(\mathbf{r}), \mathbf{r} \in \mathcal{R}\}$ et $\boldsymbol{\epsilon} = \{\epsilon(\mathbf{r}), \mathbf{r} \in \mathcal{R}\}$, la même relation s'écrit :

$$p(\mathbf{g}|\mathbf{f}) = \mathcal{N}(\mathbf{f}, v_\epsilon \mathbf{I}) \tag{7.3}$$

Puisque le but est d'obtenir une image reconstruite $\boldsymbol{f}$ segmentée en un nombre limité de régions homogènes, une variable cachée $\boldsymbol{z}$ pouvant prendre les valeurs discrètes $k \in \{1, ..., K\}$ est introduite. Cette variable $\boldsymbol{z}$ permet une classification des pixels de l'image $\boldsymbol{f}$ en K classes, ce qui conduit à une segmentation de l'image en régions $\mathcal{R}_k = \{\mathbf{r} : z(\mathbf{r}) = k\}$. Ces régions peuvent être connexes ou non connexes. Dans ce dernier cas, il faut les recomposer en régions connexes afin de conduire à la segmentation finale.
Une **seconde hypothèse** porte sur l'image $\boldsymbol{f}$ qui est supposée être composée de régions homogènes. Cette notion d'homogénéité est traduite par l'attribution d'une distribution gaussienne :

$$p(f(\mathbf{r})|z(\mathbf{r}) = k) = \mathcal{N}(m_k, v_k) \tag{7.4}$$

L'image peut alors être représentée par un loi qui est la somme des gaussiennes correspondant aux régions de classe k, multipliée par la probabilité d'appartenance de chaque région $z(r)$ à cette

classe :

$$p(f(\mathbf{r})) = \sum_{k=1}^{K} \alpha_k \mathcal{N}(m_k, v_k) \tag{7.5}$$

avec

$$\alpha_k = p(z(\mathbf{r}) = k)$$

7.3 Modélisation markovienne

La modélisation markovienne d'une image permet de traiter les étiquettes des pixels comme les variables d'un champ aléatoire de Markov. Ceci signifie que la valeur d'une étiquette peut être considérée *relativement à un système de voisinage* de ce site. Dire qu'une étiquette appartient à un champ de Markov signifie que son état ne dépend que de l'état de son voisinage. Le voisinage peut être du *premier ordre*, s'il concerne les 4 plus proches voisins ($\mathcal{V}(s) = |4|$), d'ordre deux si la distance au pixel est immédiatement supérieure (pixels diagonaux) ou d'ordre supérieur s'il fait intervenir des sites de distance supérieure (voir fig. 7.2).

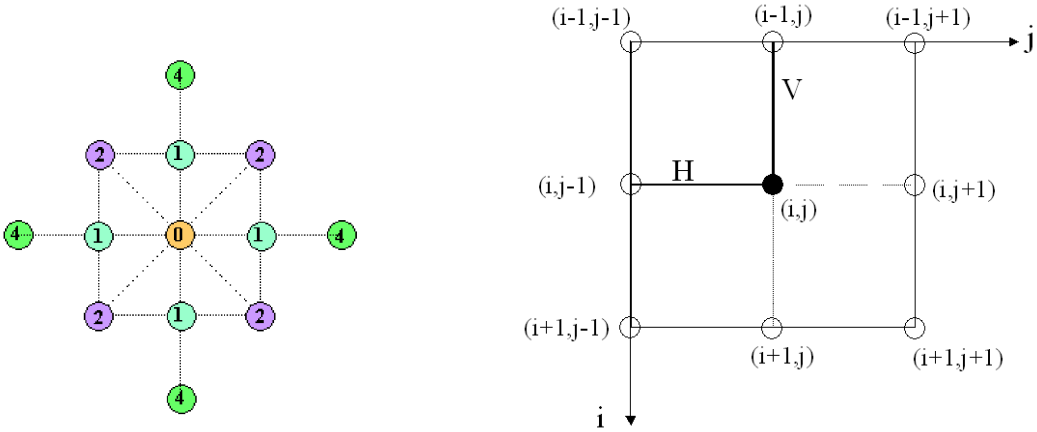


FIG. *7.2 – a) Champs de Markov d'ordre 1, 2 et 4 (d'après [Idi01]). Le voisinage d'ordre 1 est seul utilisé pour la dépendance des étiquettes dans le domaine direct des pixels de l'image. b) Dépendances d'un site (i, j) avec son voisinage d'ordre 1 (directions V, H) utilisés dans un PMRF d'ordre 1.*

7.4 Modèle de Potts-Markov

Afin de construire des régions homogènes, la dépendance spatiale entre l'étiquette de chaque pixel (la variable de Markov cachée $z(r)$) et les étiquettes des pixels voisins est modélisée par un champ de Potts-Markov (PMRF). La modélisation Markovienne suppose que la valeur de $z(r)$ à la position d'un pixel est reliée à la valeur $z(r)$ pour les pixels voisins (les quatre plus proches voisins, dans

les directions horizontale et verticale si l'on prend un voisinage du premier ordre). Le modèle de Potts permet de contrôler, au moyen d'un paramètre d'attraction-répulsion α, la valeur moyenne de la taille d'une région. Ainsi, l'homogénéité de chaque classe est proportionnelle à l'amplitude de α. L'augmentation de α permet d'obtenir des zones homogènes de plus grande taille.

$$p(z(\boldsymbol{r}), \boldsymbol{r} \in \mathcal{R}) = \frac{1}{T(\alpha)} \exp\left\{ \alpha \sum_{\boldsymbol{r} \in \mathcal{R}} \sum_{\boldsymbol{s} \in V(\boldsymbol{r})} \delta(z(\boldsymbol{r}) - z(\boldsymbol{s})) \right\} \tag{7.6}$$

où $V(\boldsymbol{r})$ représente le voisinage de $\boldsymbol{r}$. Dans la suite de ce chapitre, nous considérerons $V(\boldsymbol{r})$ comme un voisinage de premier ordre, ou 4-connexité, du pixel $\boldsymbol{r}$ (voir fig. 7.2). Dans le chapitre 9 nous ferons évoluer ce modèle du premier ordre vers un modèle combinant les ordres 1 et 2 pour s'adapter aux orientations des sous-bandes d'ondelettes.
Le modèle de Potts, lorsque l'on considère le voisinage de premier ordre d'un pixel, peut s'exprimer de façon plus explicite avec les indices (i, j) de chaque pixel $r(i, j)$:

$$\begin{aligned} p(z(i,j), (i,j) \in \mathcal{R}) = \frac{1}{T(\alpha)} \times \exp\Bigg\{ & \alpha \sum_{(i,j) \in \mathcal{R}} \delta(z(i,j) - z(i,j-1)) \\ & + \alpha \sum_{(i,j) \in \mathcal{R}} \delta(z(i,j) - z(i-1,j)) \Bigg\} \end{aligned} \tag{7.7}$$

D'après la règle de Bayes, la loi a posteriori s'écrit :

$$p(\boldsymbol{f}, \boldsymbol{z} | \boldsymbol{g}) \propto p(\boldsymbol{g} | \boldsymbol{f}, \boldsymbol{z})\, p(\boldsymbol{f} | \boldsymbol{z})\, p(\boldsymbol{z}) \tag{7.8}$$

Les lois a priori, définies précédemment, nécessitent que certains de leur paramètres, appelés *hyper-paramètres*, soient précisés. Ces paramètres sont v_ϵ, v_k et m_k. Si nous voulons réaliser une segmentation non-supervisée, l'ensemble des paramètres que nous appellerons :

$$\boldsymbol{\theta} = \left\{ v_\epsilon, (m_k, v_k), \quad k \in \{1...K\} \right\} \tag{7.9}$$

doit aussi être *estimé*. Dans ce but nous assignons aussi des lois a priori aux paramètres $\boldsymbol{\theta}$. Ce *deuxième groupe de lois a priori*, qui concerne l'estimation des hyperparamètres, est choisi en prenant des lois *a priori conjuguées* qui dépendent elle-mêmes de ce que l'on appelle les *hyper-hyperparamètres*. Ceux-ci seront appelés α_0, β_0, m_0 et v_0. Nous nous référons à [SMD02] pour le choix et les valeurs à donner à ces hyper-hyperparamètres. Les lois conjuguées choisies pour l'ensemble $\boldsymbol{\theta}$ des hyperparamètres sont définies par :

$$\left\{ \begin{array}{lcl} p(v_\epsilon) & \sim & \mathcal{IG}(v_\epsilon | \alpha_0^\epsilon, \beta_0^\epsilon) \\ p(m_k) & \sim & \mathcal{N}(m_k | m_0^k, v_0^k)\ ,\ k = \{1...K\} \\ p(v_k) & \sim & \mathcal{IG}(v_k | \alpha_0^k, \beta_0^k)\ ,\ k = \{1...K\} \end{array} \right. \tag{7.10}$$

où $\mathcal{IG}$ est la loi Inverse-Gamma.

L'expression de la nouvelle loi a posteriori pour un modèle de segmentation *non-supervisée* devient :

$$p(\boldsymbol{f}, \boldsymbol{z}, \boldsymbol{\theta} | \boldsymbol{g}) \propto p(\boldsymbol{g} | \boldsymbol{f}, \boldsymbol{z}, \boldsymbol{\theta})\, p(\boldsymbol{f} | \boldsymbol{z}, \boldsymbol{\theta})\, p(\boldsymbol{z})\, p(\boldsymbol{\theta}) \tag{7.11}$$

7.5 Approximation de Monte Carlo (MCMC) et algorithme de Gibbs

L'approche bayésienne, avec les choix faits pour notre modèle, nécessite maintenant d'estimer l'ensemble des variables $(\boldsymbol{f}, \boldsymbol{z}, \boldsymbol{\theta})$ en fonction de la distribution a posteriori exprimée dans la relation 7.11. En effet, le calcul de cette loi a posteriori est difficilement réalisable analytiquement, voire pas du tout, dans la plupart des cas de figure. Les méthodes de Monte Carlo appliquées aux chaînes de Markov (MCMC) nous apportent une heuristique qui consiste à calculer un grand nombre de réalisations, ou échantillons, de la loi a posteriori 7.11. A partir de ces *réalisations numériques*, il suffit alors de calculer une estimation de type moyenne ou médiane, ou encore "modes" (de l'histogramme des valeurs prises par chaque pixel dans l'ensemble des itérations). L'approximation de Monte Carlo, bien connue des physiciens, est une application de la *loi faible des grands nombres*. La méthode de Monte Carlo est mise en oeuvre ici en utilisant un algorithme de Gibbs.

Lors des itérations de Gibbs sont estimées les moyennes, variances et autres statistiques pour chaque variable $\boldsymbol{f}$, $\boldsymbol{z}$ et $\boldsymbol{\theta}$. Par exemple la moyenne de $\boldsymbol{f}$ devient :

$$\hat{\boldsymbol{f}} = \int \boldsymbol{f}.\, p(\boldsymbol{f}|\boldsymbol{g}) d\boldsymbol{f} \simeq \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{f}^{(n)} \tag{7.12}$$

L'algorithme d'échantillonnage de Gibbs, décrit en 7.13 ci-dessous, consiste dans le calcul successif des variables $\boldsymbol{f}$, $\boldsymbol{z}$ et $\boldsymbol{\theta}$, décrites chacune par leur loi a posteriori, et pour chaque itération $n \in \{1...itermax\}$, où $itermax$ est le nombre total de réalisations nécessaires pour que l'algorithme converge. Par convergence, nous entendons le moment, ou le nombre d'itérations, à partir duquel les valeurs des étiquettes des pixels se stabilisent autour d'une valeur moyenne et n'évoluent quasiment plus.
Algorithme d'échantillonnage de Gibbs :

$$\left\{ \begin{array}{lcl} \boldsymbol{f}^n & \sim & p(\boldsymbol{f}|\boldsymbol{g}, \boldsymbol{z}^{(n-1)}, \boldsymbol{\theta}^{(n-1)}) \\ \boldsymbol{z}^n & \sim & p(\boldsymbol{z}|\boldsymbol{g}, \boldsymbol{\theta}^{(n-1)}, \boldsymbol{f}^{(n-1)}) \\ \boldsymbol{\theta}^n & \sim & p(\boldsymbol{\theta}|\boldsymbol{g}, \boldsymbol{z}^{(n-1)}, \boldsymbol{f}^{(n-1)}) \end{array} \right. \tag{7.13}$$

Nous avons introduit dans cet algorithme l'expression de trois nouvelles lois a posteriori pour $\boldsymbol{f}$, $\boldsymbol{z}$ et $\boldsymbol{\theta}$. Nous allons maintenant écrire les expressions de ces lois.

1) pour $\boldsymbol{f}$

$$\begin{array}{rcl} p(\boldsymbol{f}|\boldsymbol{g}, \boldsymbol{z}, \boldsymbol{\theta}) & \propto & p(\boldsymbol{g}|\boldsymbol{f}, \boldsymbol{z}, \boldsymbol{\theta})\; p(\boldsymbol{f}|\boldsymbol{z}, \boldsymbol{\theta}) \\ & \propto & \prod_k p(\boldsymbol{g}_k|\boldsymbol{f}_k, v_k)\; p(\boldsymbol{f}_k|m_k, v_k) \\ & \propto & \prod_k \prod_{\boldsymbol{r} \in R_k} \mathcal{N}(g(\boldsymbol{r})|f(\boldsymbol{r}), v_\epsilon) \mathcal{N}(f(\boldsymbol{r})|m_k, v_k) \\ & \propto & \prod_k \prod_{\boldsymbol{r} \in R_k} \mathcal{N}(f(\boldsymbol{r})|\hat{m}_k, \hat{v}_k) \end{array} \tag{7.14}$$

où

$$\hat{m_k} = \hat{v}_k \left(\frac{m_k}{v_k} + \frac{\sum\limits_{\boldsymbol{r} \in \mathcal{R}_k} g(\boldsymbol{r})}{v_\epsilon} \right) \qquad \text{et} \qquad \hat{v}_k = \left(\frac{1}{v_\epsilon} + \frac{1}{v_k} \right)^{-1} \tag{7.15}$$

et

$$R_k = \{\boldsymbol{r} : z(\boldsymbol{r}) = k\} \tag{7.16}$$

2) pour $\boldsymbol{z}$

$$\begin{aligned} p(\boldsymbol{z}|\boldsymbol{g}, \boldsymbol{f}, \boldsymbol{\theta}) &\propto p(\boldsymbol{g}|\boldsymbol{f}, \boldsymbol{z}, \boldsymbol{\theta})\, p(\boldsymbol{f}|\boldsymbol{z}, \boldsymbol{\theta})\, p(\boldsymbol{z}|\boldsymbol{\theta}) \\ &\propto \left[\prod_k p(\boldsymbol{g}_k|\boldsymbol{f}_k, v_\epsilon)\, p(\boldsymbol{f}_k|m_k, v_k) \right] p(\boldsymbol{z}) \\ &\propto \left[\prod_k \prod_{\boldsymbol{r} \in R_k} \mathcal{N}(f(\boldsymbol{r})|\hat{m_k}, \hat{v_k}) \right] p(\boldsymbol{z}) \end{aligned} \tag{7.17}$$

Nous pouvons noter que cette loi a posteriori est aussi un champ de Potts-Markov où les probabilités a priori sont pondérées par la vraisemblance a posteriori.

3) pour $\boldsymbol{\theta}$

$$p(\boldsymbol{\theta}|\boldsymbol{f}, \boldsymbol{g}, \boldsymbol{z}) \propto p(v_\epsilon|\boldsymbol{f}, \boldsymbol{g}) \prod_k p(m_k|v_k, \boldsymbol{f}, \boldsymbol{z}).p(v_k|\boldsymbol{f}, \boldsymbol{z}) \tag{7.18}$$

où :
- Pour la variance de bruit v_ϵ :

$$p(v_\epsilon|\boldsymbol{f}, \boldsymbol{g}) \propto \mathcal{IG}(v_\epsilon, |\alpha, \beta) \tag{7.19}$$

avec

$$\alpha = n/2 + \alpha_0^\epsilon \qquad \text{et} \qquad \beta = \frac{1}{2} \sum_{\boldsymbol{r} \in \mathcal{R}} (g(\boldsymbol{r}) - f(\boldsymbol{r}))^2 + \beta_0^\epsilon$$

où $n = Card(\mathcal{R})$.

- Pour la moyenne m_k dans chaque région k :

$$p(m_k|\boldsymbol{f}, \boldsymbol{z}, v_k, m_0, v_0) \propto \mathcal{N}(m_k|\mu_k, \xi_k) \tag{7.20}$$

où

$$\mu_k = \xi_k \left(\frac{m_0^k}{v_0^k} + \frac{\sum\limits_{\boldsymbol{r} \in \mathcal{R}_k} f(\boldsymbol{r})}{v_k} \right), \qquad \xi_k = \left(\frac{n_k}{v_k} + \frac{1}{v_0^k} \right)^{-1} \qquad \text{et} \qquad n_k = Card(R_k)$$

- Pour la variance v_k dans chaque région k :

$$p(v_k) \propto \mathcal{IG}(v_k|\alpha_k, \beta_k) \tag{7.21}$$

avec

$$\alpha_k = \alpha_0^k + \frac{n_k}{2} \qquad \text{et} \qquad \beta_0^k = \beta_0 + \frac{1}{2}\sum_{r\in\mathcal{R}_k}(f(\boldsymbol{r}) - m_k)^2$$

Nous avons précisé que l'algorithme de Gibbs est itéré un nombre "suffisant" de fois ($itermax$) afin d'atteindre la convergence, c'est-à-dire un état stable de la segmentation et des paramètres des modèles utilisées. Nous n'utilisons pas de réel "critère de convergence". Une fois le nombre $itermax$ atteint, nous ne conservons que les *valeurs significatives de l'échantillonnage* afin d'obtenir le résultat de la segmentation $\boldsymbol{f}$. En effet, si nous engendrons un nombre d'échantillons $itermax = N$,

$$(\boldsymbol{f}, \boldsymbol{z}, \boldsymbol{\theta})^{(1)}, (\boldsymbol{f}, \boldsymbol{z}, \boldsymbol{\theta})^{(2)}, ...(\boldsymbol{f}, \boldsymbol{z}, \boldsymbol{\theta})^{(L)}, ..., (\boldsymbol{f}, \boldsymbol{z}, \boldsymbol{\theta})^{(N)}$$

l'algorithme ne commence à fournir des valeurs significatives de la segmentation, et des paramètres, qu'au bout d'un "temps de chauffe" que l'on estime à environ 30% du nombre total N d'itérations. Soit L le nombre d'échantillons correspondant à ce temps de chauffe, on n'utilisera donc que les $N-L$ valeurs restant pour le calcul de l'estimé. Si nous choisissons comme estimateur le "mode", qui correspond à la valeur de l'étiquette d'un pixel la plus fréquemment rencontrée lors des itérations, ce mode sera donc calculé sur les $N - L$ dernières itérations. La valeur finale pour chaque pixel sera donnée par :

$$(\hat{\boldsymbol{f}}, \hat{\boldsymbol{z}}, \hat{\boldsymbol{\theta}}) \simeq \arg\max hist_{N-L}(\boldsymbol{f}, \boldsymbol{z}, \boldsymbol{\theta})\} \tag{7.22}$$

Notre expérience des images analysées a montré que $itermax$ dépend essentiellement de la complexité de l'image et du nombre de niveaux de luminance dans l'image (donc de régions homogènes) ainsi que du nombre de classes demandées à la segmentation. En général, pour un nombre de classes $K = 4$, nous prenons $itermax$ de l'ordre de 100.

7.6 Parallélisation de l'échantillonneur de Gibbs

Nous avons vu que l'échantillonneur de Gibbs nécessite, dans un système à K variables, d'échantillonner un paramètre à la fois pour réaliser une iteration complète. Notre image est un système à $M \times N$ pixels et nécessiterait donc d'effectuer autant d'échantillonnages successifs pour réaliser une itération complète de l'échantillonneur décrit en 7.13. Si l'on prend soin de remarquer que le modèle de voisinage utilisé est d'ordre 1, une solution pour diminuer le nombre d'itérations consiste à séparer l'image en deux sous-ensembles de sites indépendants : les sites correspondants aux cases blanches (sites impairs) d'un échiquier et les sites correspondant aux cases noires (sites pairs). En effet, si nous considérons que la valeur de l'étiquette d'un site est conditionnelle uniquement aux étiquettes des sites de son voisinage d'ordre 1 ($p(z(i,j)|z(i,j-1), z(i-1,j))$, alors tous les sites "blancs" peuvent être considérés comme indépendants connaissant les étiquettes des sites noirs. Réciproquement, une fois les étiquettes des sites blancs connus, les sites noirs peuvent être considérés comme indépendants. Cette remarque faite, on est donc en mesure d'effectuer une itération complète des sites de l'image en deux échantillonnages successifs sur deux sous-images correspondant aux

sites noirs et blancs d'un échiquier. Cette méthode de parallélisation de l'échantillonnage permet de réduire le nombre d'itérations sur la variable z, de $M \times N$, taille de l'image, à seulement deux.

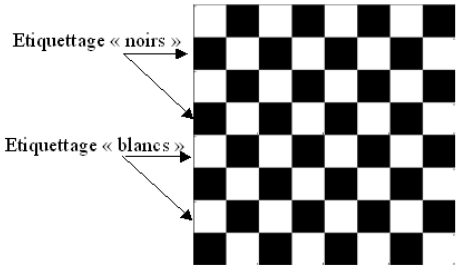


FIG. *7.3 – Répartition en deux sous-ensembles de sites blancs et noirs, sous la forme d'un échiquier, pour la mise en oeuvre, en parallèle, "en deux coups" d'une itération de l'échantillonneur de Gibbs sur l'image totale. Le voisinage considéré est un voisinage d'ordre 1 (directions horizontale et verticale). Les sites d'un même sous-ensemble (noirs ou blancs) peuvent être considérés comme indépendants conditionnellement à la connaissance de l'autre sous-ensemble (respectivement blancs ou noirs). Ceci permet l'échantillonnage en un coup de tous les sites d'un même type (blanc par exemple) puis de tous les sites de l'autre type (noirs).*

7.7 Conclusion au chapitre 7

Nous avons présenté dans ce chapitre le principe de la segmentation par approche bayésienne non supervisée et modélisation markovienne dans le domaine direct de l'image (BPMS : Bayesian Potts-Markov Segmentation). Cette méthode utilise un modèle de Potts-Markov pour les étiquettes des pixels et repose sur une hypothèse d'indépendance des pixels dans chaque région. Dans les deux chapitres qui suivent nous présentons les travaux que nous avons réalisés sur la base de cette méthode. Le premier de ces travaux est une application directe et simple de cette méthode et concerne la segmentation rapide de séquences vidéo. Le deuxième concerne l'amélioration de la vitesse de segmentation sur des images fixes. Il consiste à réaliser tout d'abord une projection de l'image observée dans le domaine des coefficients d'ondelette. Puis la méthode de segmentation présentée dans le domaine direct est appliquée dans le domaine des coefficients d'ondelette de l'image. Quelques modifications à la démarche initiale ont été apportées pour pouvoir appliquer cette méthode et le modèle de Potts a été étendu aux directions privilégiées des sous-bandes d'ondelettes. Les résultats comparatifs entre la segmentation dans le domaine direct et la méthode que nous avons développée dans le domaine des ondelettes seront discutés dans le chapitre 10.

Chapitre 8

Segmentation bayésienne de séquences vidéo dans le domaine direct

Nous avons expliqué, dans le chapitre précédent, comment mettre en oeuvre une méthode bayésienne de résolution d'un problème inverse : celui de la segmentation d'images fixes. Nous avons montré les résultats obtenus sur des images fixes et argumenté le choix de nos modèles, des algorithmes ainsi que des paramètres de la segmentation (paramètre de réglage du champ de Potts, choix de l'initialisation pour l'ensemble des inconnues, nombre d'itérations en fonction du nombre de classes). Revenons sur la possibilité d'initialiser notre algorithme de segmentation par une connaissance a priori de cette segmentation. Nous pouvons rencontrer cette possibilité dans le cas particulier de déplacements relativement faibles entre des images. C'est en général le cas dans une même scène d'une séquence vidéo. Notre méthode bayésienne de segmentation peut alors être "accélérée" pour la segmentation de l'ensemble d'une même scène en tenant compte de cette remarque.
L'application de notre méthode de segmentation bayésienne, avec champ de Potts, a donc été testée sur des séquences vidéo de la façon suivante : si l'on considère que les images successives (ou frames) d'une séquence vidéo présentent des variations faibles entre elles, le résultat de la segmentation de la première image d'une séquence peut être re-utilisé pour initialiser la segmentation de l'image suivante. Ce principe permet de réduire considérablement le nombre d'itérations pour atteindre la convergence sur l'image No 2. Si l'on propage ce principe à toutes les images de la séquence, il n'est plus nécessaire de segmenter, avec un nombre élevé d'itérations, que la première image de la séquence. Ce principe, très simple, nous permet de diminuer le nombre d'itérations de l'algorithme d'échantillonnage de Gibbs pour segmenter toutes les $N-1$ images, après la première, dans une même scène.
Le modèle de segmentation d'images fixes est donc facilement applicable à la segmentation de séquences "$2D+T$", que l'échantillonnage soit régulier (vidéo) ou non. L'originalité réside ici dans le fait que la segmentation de séquences est faite de façon incrémentale d'une trame (image) à la suivante dans un champ de Markov "temporel". De cette façon, nous avons démontré [5] une amélioration significative de la vitesse de segmentation d'une séquence. Nous avons aussi montré l'intérêt de ce résultat pour la détection et l'estimation de mouvement ainsi que pour la compression de séquences.

La suite de ce chapitre est organisée de la façon suivante :
- La section 8.1 présente brièvement l'extension de la méthode de segmentation bayésienne, dans le domaine direct, à la segmentation de séquences $2D + T$, en explicitant le choix des paramètres d'initialisation de segmentation.
- La section 8.2 présente le résultat de segmentation sur une séquence de 30 trames comportant des mouvements locaux et un mouvement global (séquence "joueur de tennis" dont le mouvement global est un zoom arrière en fin de séquence).
- La section 8.3 présente et commente les résultats de cette approche de la segmentation par rapport à d'autres techniques de segmentation qui conduisent à une compression basée sur la quantification.
- La section 8.1 résume l'extension de la méthode bayésienne dans le domaine direct à la segmentation de séquences $2D + T$.
- La section 8.4 montre comment utiliser les résultats de la segmentation 2D+T pour obtenir une quantification des paramètres de mouvement de différents objets dans une scène vidéo.

8.1 Algorithme de segmentation de séquences

Une séquence vidéo peut être représentée par une série d'image fixes, appelées "trames" (frames) et présentant une relation entre les objets appartenant à deux trames successives que l'on nomme "mouvement". Nous faisons considérerons ici que le mouvement entre trames est considéré dans le cas d'un balayage non-entrelacé c'est-à-dire progressif de façon simplifier la modélisation du mouvement dans la succession des trames. Ce mouvement est bien sûr en relation étroite avec la notion de temps et particulièrement avec celle d'intervalles de temps successifs identiques (période d'échantillonnage de trame ou "frame rate" de la vidéo). Ces séquences sont donc nommées $2D+T$ pour les distinguer de *séquences 3D* (et non d'images 3D) où aucune relation de temps précise n'existe entre les images 2D. On pourrait étendre ceci à des *modèles spatiaux-tridimensionnels + temps* que l'on appellerait $4D$ ou $3D + T$, selon qu'ils sont ou non échantillonnés de façon précise dans le temps.
L'algorithme de segmentation d'une séquence vidéo est décrit par le étapes suivantes :

1) Sélection du nombre K d'étiquettes (ou de classes). Pour la séquence Tennis nous avons choisi une classification en $K = 4$ classes. Ce choix est motivé par le fait que la séquence comporte peu de régions de différente nature. Pour des séquences plus complexes nous pourrions prendre K plus élevé et fonction du nombre apparent de régions distinctes dans la séquence.
2) Nous réalisons la segmentation de la première trame de la séquence en prenant un nombre élevé d'itérations. Cette première segmentation s'effectue avec une initialisation aléatoire $Z_0(r)$, car nous n'avons aucune information a priori sur la valeur des $Z_1(r)$, la segmentation de cette 1ère image. La convergence est donc atteinte plus lentement que pour des trames dont on possède déjà une initialisation approchée de la segmentation finale.
3) La trame N2 est segmentée en utilisant comme initialisation la segmentation $Z_1(r)$ obtenue pour la trame N1.
4) Toutes les trames f_n, $n = \{2...N\}$, sont segmentées sur la base de la segmentation $Z_{n-1}(r)$

obtenue à la trame précédente f_{n-1}. La connaissance de $Z_{n-1}(r)$ permet de réduire le nombre d'itérations entre trames à une valeur très basse (par exemple $itermax = 6$).

8.2 Exemple de segmentation de séquence

Nous présentons nos résultats de segmentation sur une partie de la séquence "Tennis". Cette séquence est composée de 30 trames de taille 320×240 sur 8 bits. Nous sélectionnons une partie de la séquence composée de 6 trames. Nous montrons ensuite le résultat de la segmentation en 4 classes sur la première trame, puis le résultat de la segmentation sur les cinq autres trames de ce GOF. Le nombre d'itérations pour segmenter la première image est de 20. Pour les images suivantes, le nombre d'itérations est réduit à 6. Afin de montrer comment évolue la segmentation au cours de l'échantillonnage de Gibbs entre deux trames, nous affichons, dans la figure 8.1, la segmentation de la trame après chacune des itérations.

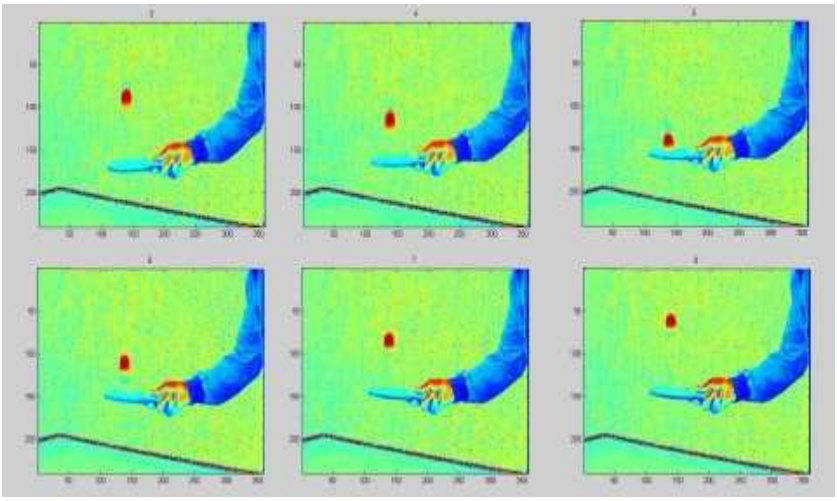


FIG. *8.1 – La séquence originale "Tennis", réduite à 6 images, sur laquelle nous avons testé notre méthode de segmentation de séquence par initialisations successives. Les images sont de taille* 320×240 *sur 8 bits.*

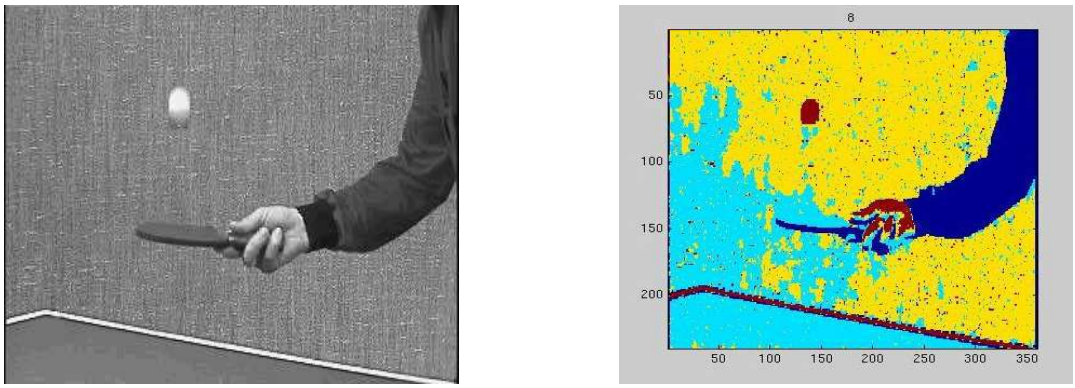


FIG. *8.2 – a) Une des 6 images de la séquence "joueur de tennis". b) Segmentation de l'image en 4 classes.*

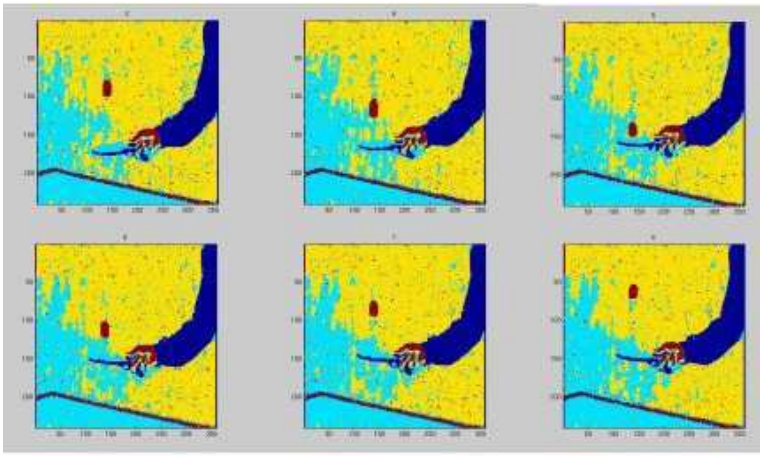


FIG. *8.3 – Segmentation bayésienne "directe", en $K = 4$ classes, des 6 images de la séquence "joueur de tennis" : La trame initiale est segmentée en $= 20$ itérations. Les 5 trames suivantes sont segmentées avec initialisation par le résultat de la segmentation de la trame précédente et seulement 6 itérations.*

8.3 Compression post-segmentation

La segmentation de séquence selon cette méthode permet une compression de la séquence d'un facteur correspondant à la réduction du nombre de bits de quantification nécessaires. Dans le cas de quatre classes, on passe donc de huit à deux bits, ce qui représente un taux de compression

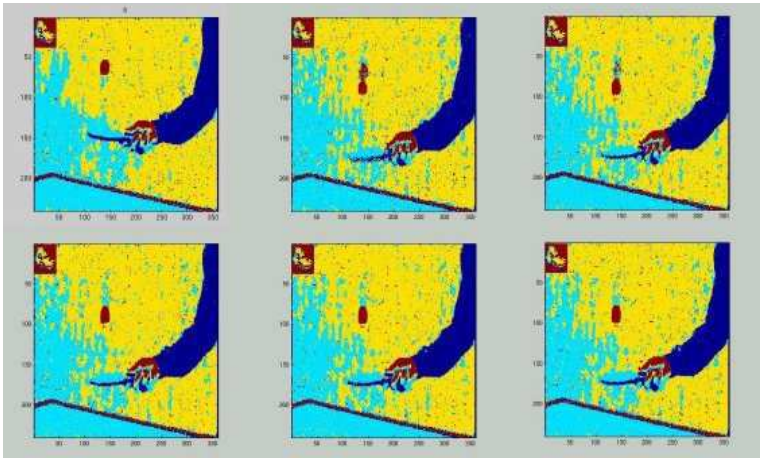


FIG. *8.4 – Cette figure montre le comportement de l'algorithme itératif (6 itérations) entre deux images successives. En haut à gauche, l'image initiale ; puis, toujours de gauche à droite et de haut en bas, les* 5 *itérations successives conduisant à la segmentation de l'image suivante. On peut noter que la segmentation finale est déjà atteinte à la* 3*ème itération (les segmentations des itérations* 4, 5 *et* 6 *sont identiques). Ceci signifie que la segmentation pourrait, pour cette vitesse de déplacement, se faire en deux fois moins d'itérations.*

de 4 sur la séquence. Cette compression, que l'on estime difficile à réaliser "en temps réel", ne doit cependant pas être négligée car elle représente simultanément une réduction de la complexité de la séquence tout en effectuant une segmentation utilisable pour l'estimation de mouvement comme nous allons le montrer. De plus les temps de segmentation, très réduits grâce à la méthode d'initialisation de la segmentation, sont aussi très relatifs aux machines sur lesquelles nous les avons testés et ne sont certainement pas optimisés. En revanche, la qualité de la segmentation est bien supérieure à celle d'un simple "partitionnement" de l'image, comme celle d'un modèle d'arbre quaternaire (quadtree) par exemple.

8.4 Estimation de mouvement post-segmentation

Une fois réalisée la segmentation en régions homogènes d'une séquence $2D + T$, nous sommes en mesure de calculer facilement les paramètres cinématiques relatifs à des régions ou des "objets" de la séquence. Cette approche "contextuelle" est ici très différente des approches à "champ dense de mouvement", telles que le flot optique. Elle s'appuie sur une "forme basique" de reconnaissance des objets en classes. Considérons notre séquence "joueur de tennis" segmentée en $K = 4$ classes. Si nous voulons quantifier le mouvement de la balle de tennis entre deux trames, nous allons considérer uniquement les images $z_k(r)$ avec $k \in \{1...K\}$, où k est la classe à laquelle appartient la balle de tennis. Nous considérons donc uniquement les objets qui appartiennent à cette classe. Ceci revient

à simplifier l'image, segmentée en 4 classes, en une image binaire où toutes les régions de la classe k sont numérotées en série, selon le voisinage choisi (ordre 1).
Ensuite, nous attribuons à chaque région un numéro d'étiquette, ce qui assigne à chaque pixel de cette région le même numéro. Nous calculons la masse de l'objet d'intérêt et son centroïde (barycentre) en utilisant les numéros d'étiquette attribués aux pixels de cet objet. Nous répétons la même opération pour la trame suivante. Nous recherchons alors les objets de classe k entre la trame n et la trame $n+1$, contenus dans un voisinage ou "fenêtre" de mouvement et qui possèdent une masse proche de la masse de l'objet initial (une différence de masse inférieure ou égale à 5% pour l'objet suivi). Ceci permet de suivre de façon non-supervisée l'objet d'intérêt.
Il est alors facile de calculer le vecteur de mouvement de tout l'objet, entre deux trames successives, en calculant la distance entre les centroïdes de l'objet dans les trames n et $n+1$.
Ceci est montré dans la figure 8.6 où les centroïdes $B1(X_1, Y_1)$ et $B2(X_2, Y_2)$ sont représentés par un cercle au milieu de la balle dans les trames n et $n+1$. Dans l'exemple montré ici nous pouvons donner avec précision la vitesse $\vec{X} = 1$ en pixels/trame sur l'axe Ox et $\vec{Y} = 16$ en pixels/trame sur l'axe Oy.
Pour des mouvements importants, il peut être intéressant d'utiliser toutes les itérations de l'algorithme de Gibbs entre 2 trames, dans le but de restreindre le déplacement entre deux itérations et de faciliter le suivi de l'objet. Cette amélioration n'a pas encore été mise au point, mais elle permettrait probablement de répondre non seulement à des changements de paramètres cinématiques importants (vitesse, dérivée d'ordre 2 par rapport à x ou y) de forme, et à des transformations linéaires ou non. On pourrait alors de se replacer dans le cadre évoqué dans la première partie où les paramètres de déplacement des objets permettraient de construire, de suivre et de modéliser la trajectoire de l'objet dans la séquence.
Une compression, basée non plus sur de simples vecteurs de mouvement par blocs mais sur la connaissance des paramètres de mouvement attribués à des objets distincts, pourrait ainsi être réalisée. Comme nous l'avons déjà proposé dans la première partie, les paramètres de mouvement permettent alors d'identifier le mouvement des objets à des "trajectoires" et la prédiction du mouvement est basée sur la connaissance de cette trajectoire. Là encore nous pensons qu'il est intéressant de réaliser une EM basée davantage sur l'aspect de la compréhension de la scène plutôt que sur le codage brut de vecteurs de mouvement attribués à des blocs de l'image, vecteurs dont la signification est sans aucun rapport avec la compréhension de la scène.

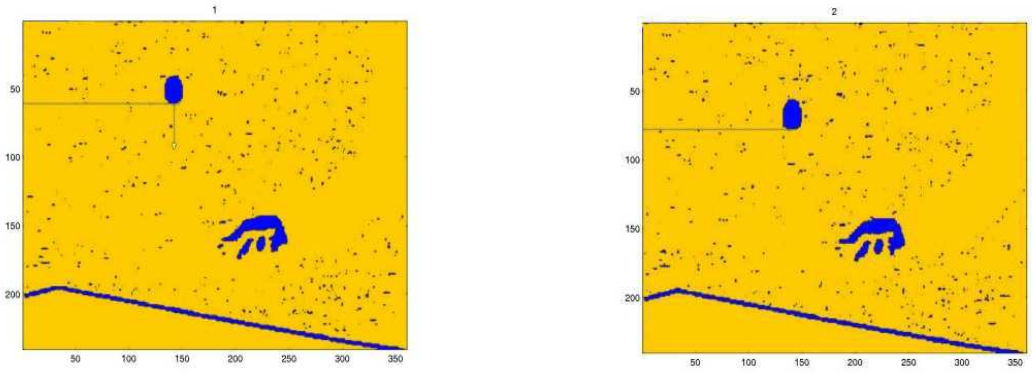


FIG. *8.5 – Images binaires des deux trames successives montrant les régions de classe* 4*, dont fait partie la balle a) Image binaire obtenue par segmentation de la trame* 1*. b) Image binaire obtenue à partir de la trame No* 2 *suivante.*

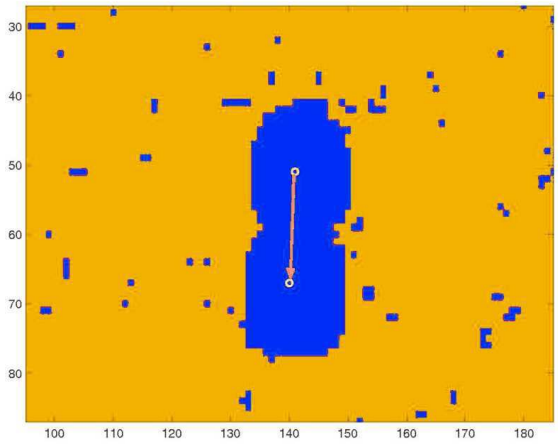


FIG. *8.6 – On montre ici les trames* 1 *et* 2 *superposées, avec un grossissement sur la balle. Le centroïde de la balle est repéré par chaque cercle et la flèche matérialise le vecteur de mouvement entre les trames* 1 *et* 2*.*

8.5 Conclusion sur la segmentation de séquences

Nous avons proposé l'adaptation d'une méthode de segmentation bayésienne, fondée sur une approche de Potts-Markov, à la segmentation de séquences vidéo (espace 2D+T). L'intérêt d'une telle méthode est multiple :

- Le faible nombre de classes par lequel est caractérisée chaque image de la séquence permet de quantifier une image sur 8 bits en seulement 2 bits, tout en conservant une bonne homogénéité entre les régions. Ceci montre la possibilité de réaliser une compression de la séquence vidéo au moyen d'une segmentation totalement non-supervisée.

- L'analyse du mouvement de régions appartenant à la même classe, dans les trames segmentées, est utilisée pour l'estimation du mouvement de régions ou d'objets spécifiques. Ceci est réalisé de façon totalement non-supervisée en utilisant le déplacement du centre de masse d'un objet (ou d'une région) dans une "fenêtre de mouvement" limitée. Cette fenêtre correspond à un voisinage spatial limité autour de l'objet. Ce principe est classique en vidéo dans les approches de compensation de mouvement par mise en correspondance de blocs (BM ou block-matching), dans le sens où l'on recherche dans une trame $n+1$, et dans un voisinage limité autour d'un bloc $B(i,j)$ précis de la trame n, le bloc de luminance moyenne la plus proche de celle du bloc $B(i,j)$.

- Le cadre de nos applications n'est pas nécessairement celui du temps réel dans la mesure où l'on considère que les séquences que nous avons à traiter peuvent l'être soit en temps légèrement différé, soit après acquisition complète et traitement "off line". En revanche, nous nous intéressons à une segmentation de bonne qualité et au fait que les résultats sur l'estimation et la modélisation du mouvement sont obtenus de façon totalement non-supervisée. En particulier nous avons commencé, dans le cadre d'une action concertée avec le LIP6 et une unité INSERM U538 du CHU St Antoine, à nous intéresser au traitement de séquences vidéo de phénomènes biocellulaires qui se déroulent sur de grandes échelles de temps. Ces phénomènes biocellulaires, brièvement présentés, sont caractérisés par le déplacement lent d'un élément, appelé "raft", à l'intérieur d'une vésicule, lorsqu'un organisme est soumis à une agression de nature biologique. L'idée, dans ce cadre, est de réaliser une mesure quantitative du phénomène dynamique physique par analyse de vitesse, de trajectoire et de déformation. De plus le système d'acquisition produisant une masse importante de données vidéo, l'automatisation de certaines phases de l'analyse, dont le démarrage automatique d'un enregistrement couplé à la détection d'un déplacement du raft, est fortement envisagée et abordable par la méthode que nous venons de présenter.
- Enfin, nous verrons que cette approche d'estimation spatio-temporelle du mouvement basée sur une segmentation "itérative" peut être améliorée en utilisant une nouvelle méthode de segmentation dans le domaine des ondelettes, méthode qui est décrite dans le chapitre 9 qui suit. La méthode d'estimation ainsi réalisée est en quelque sorte une réciproque de la méthode d'estimation de mouvement présentée dans la première partie. Elle nous permet de réaliser une EM basée segmentation alors que la TO adaptée mouvement regroupe les objets possédant les mêmes caractéristiques cinématiques et se comporte donc comme une segmentation basée mouvement.

Chapitre 9

Segmentation bayésienne d'images et de séquences dans le domaine ondelettes

La segmentation bayésienne présentée jusqu'ici peut aussi être développée dans un domaine transposé du domaine direct. L'intérêt de cette transposition réside alors dans l'adjonction de propriétés particulières des "sites" dans le domaine transformé. Cette transformation peut être une transformée de Fourier, une transformée de Hadamard ou encore une transformée en ondelettes. Le GPI s'est déjà intéressé au domaine des ondelettes pour des applications de séparation de sources [IMD04], et le choix s'est porté aussi sur cette transformation pour la réalisation de notre modèle de segmentation d'images.
Une grande partie du développement de ce chapitre, ainsi que de ses résultats, sont extraits de la référence [2]. Ces résultats, récents, ont été mis au point à l'occasion de cette publication et ont été obtenus simultanément à l'écriture de ce chapitre.

9.1 Etat de l'art en segmentation dans le domaine des ondelettes

La segmentation supervisée, partiellement supervisée ou totalement non-supervisée d'images fixes a été déjà étudiée et largement décrite dans la littérature [SF02].
Plusieurs développements de la segmentation bayésienne [CB01b, CNB98, RCB01, SF02] basés sur les décompositions multi-échelle, dans le domaine des ondelettes et par "arbre caché de Markov" (HMT : Hidden Markov Tree), utilisent également cette propriété intéressante des ondelettes de présenter des coefficients significatifs très "épars" (sparse), ce qui permet de réduire la complexité de calcul.

Au GPI, une méthode de segmentation entièrement non-supervisée a été développée dans le domaine direct. Cette méthode, décrite au chapitre 7, exploite un modèle de Markov caché (HMM), le modèle de Potts-Markov. Il permet de prendre en compte la dépendance des étiquettes de pixels

immédiatement voisins (à l'ordre 1), c'est-à-dire exploite la probabilité que les étiquettes des pixels placés dans un voisinage d'ordre 1 (les quatre plus proches dans les directions horizontale H et verticale V) d'un site $s(i,j)$ prennent la même valeur que celle du site s. Dans la direction horizontale, les 2 pixels voisins seront les sites $s(i-1,j)$ et $s(i+1,j)$. Dans la direction verticale, ce seront les sites $s(i,j-1)$ et $s(i,j+1)$.
Cette dépendance est représentée par un terme d'énergie potentielle entre le pixel et son voisin, énergie à laquelle on affecte un paramètre α de dépendance qui permet de sélectionner l'importance que l'on donne à ce potentiel.
Plus précisément, on somme les énergies potentielles calculées pour l'ensemble des sites $s(i,j)$.
L'augmentation du paramètre α permet d'accroître la dépendance entre états des pixels voisins, et ainsi de rendre les régions segmentées plus ou moins homogènes et de taille variable.

Dans l'approche de la segmentation bayésienne utilisant un PMRF dans le domaine direct, le voisinage est choisi d'ordre 1. Nous avons montré, dans le chapitre précédent, que ce modèle pouvait être utilisé pour réaliser une segmentation "incrémentale" de séquences vidéo $2D+T$. Cette méthode simple a permis d'accélérer la vitesse de segmentation des séquences . Nous avons aussi montré que l'on pouvait utiliser cette segmentation, orientée région, donc orientée objet, pour la détection et la quantification du mouvement [5]. L'utilisation de cette segmentation incrémentale pour la compression de séquences a été démontrée sur un exemple simple et présente, avec l'estimation du mouvement, un premier résultat intéressant parmi les méthodes actuellement développées.

Néanmoins, ces méthodes de segmentation sont efficaces pour la segmentation de processus à longue dépendance temporelle. Nous avons donc naturellement cherché à réduire le temps de segmentation que présente la méthode de Potts-Markov dans le domaine direct. Ceci a motivé une nouvelle approche basée sur l'utilisation du domaine transformé des ondelettes pour réaliser la segmentation.

Une des propriétés des coefficients de la décomposition en ondelettes est de présenter, dans les sous-bandes de détail, une décroissance locale rapide ("fast local decay") pour un faible nombre de coefficients. Ces coefficients sont représentatifs des singularités et des régions à fort gradient de l'image. Ce sont en général des coefficients qui représentent bien les différents "contours multi-échelles" des objets dans l'image. A l'inverse, les faibles coefficients d'ondelettes, en grand nombre, sont représentatifs de petites variations locales rapides que l'on peut bien souvent apparenter à un faible bruit et qui permettent, par élimination, de faire de la représentation en ondelettes une représentation très "creuse" d'une image.

Grâce à cette propriété de décroissance locale rapide, les coefficients d'ondelette, obtenus par une décomposition orthogonale, se comportent comme un mélange de 2 gaussiennes indépendantes centrées et à moyenne nulle (IGMM : Independent Gaussian Mixture Model) [CNB98, RCB01]. Une première distribution, à variance élevée, est représentative de peu de coefficients de forte amplitude et d'importance majeure. Une seconde distribution "piquée" et à faible variance est représentative d'un grand nombre de coefficients de faible amplitude et de faible importance. Cette propriété montre que la segmentation des sous-bandes de coefficients peut être réalisée en seule-

ment 2 classes. De plus, en annulant les coefficients appartenant à la classe des coefficients de faible importance, que nous appellerons classe $k = 1$, nous effectuons un débruitage des données et augmentons la rapidité de la segmentation.

Dans cette nouvelle approche basée sur la transposition de l'image d'origine dans le domaine des ondelettes, nous avons trouvé pertinent d'améliorer le modèle de Potts-Markov pour l'adapter aux directions privilégiées des sous-bandes de détail. En effet, les sous-bandes d'ondelettes, dans une décomposition orthogonale, présentent 3 directions privilégiées : horizontale, verticale et diagonale. Ces directions privilégiées, qui correspondent à l'utilisation des filtres ondelettes $\psi\varphi$ pour la sous-bande de détails verticaux, $\psi\psi$ pour la sous-bande diagonale et $\varphi\psi$ pour la sous-bande horizontale, nous sont apparues comme moins bien exploitées par le modèle de Potts à deux directions horizontale et verticale. Il nous a donc paru intéressant d'étendre le voisinage, utilisé pour la détermination du champ de Potts, à l'ordre 1 + l'ordre 2, ce qui revient à utiliser un voisinage de type 8-connexité (voir Fig. 9.4). Nous avons donc développé le modèle de Potts de façon à prendre en compte indépendamment les directions H, V, $D1$ et $D2$, où $D1 + D2$ correspond à la sous-bande diagonale d'ondelettes et aux axes $\pi/4$ et $3\pi/4$. A chacune des 4 directions privilégiées du voisinage d'ordre $1 + 2$, nous associons aussi des valeurs particulières du paramètre α qui sont α_V, α_H, α_{D1} et α_{D2}. Le modèle de Potts devient alors, dans le cas des ondelettes orthogonales, un modèle à 4 paramètres.

9.2 Nécessité de l'utilisation d'un domaine transformé

La méthode de segmentation bayésienne, présentée dans le chapitre 7 et basée sur une modélisation par champ aléatoire de Potts-Markov, dans le domaine direct, donne de bons résultats de classification. Comme nous l'avons vu, cette méthode présente le double intérêt de fonctionner de façon totalement non-supervisée et de donner des résultats de segmentation de très bonne qualité sur des images homogènes par région et présentant des bruits de variance et de moyenne très variables selon les zones.
Néanmoins, la méthode de segmentation bayésienne dans le domaine direct utilise un échantillonnage itératif de Gibbs, dont le principal inconvénient est de présenter des temps de calcul excessifs pour certaines applications, notamment de compression vidéo, d'estimation de mouvement et de segmentation/classification rapide. L'un de nos objectifs est précisément de réaliser la segmentation de séquences d'images et nous avons cherché le moyen de réduire de façon drastique les temps de calcul obtenus par cette méthode bayésienne et champ de Potts-Markov caché.
Une heuristique relativement classique pour mettre en évidence certaines caractéristiques particulières d'un ensemble de données est de le projeter dans un espace transformé qui peut mettre en valeur ces caractéristiques. L'intérêt d'une transformation est aussi de donner une représentation beaucoup plus simple ou encore plus "creuse" des données ce qui permet de réduire immédiatement la complexité d'un algorithme. Les transformations les plus connues à notre disposition sont les transformées de Fourier, de Fourier à fenêtre, de Wigner-Ville, de Hadamard, de Kahrunen-Loewe (ou décomposition en valeurs singulières (SVD)), en composantes indépendantes (ICA), en ondelettes, en paquets d'ondelettes. Nous nous arrêterons ici car les décompositions temps-fréquence et

temps-échelle sont nombreuses.

9.3 Propriétés statistiques des coefficients d'ondelette

Pour résoudre notre problème, nous nous sommes intéressés de près aux propriétés de la transformée en ondelettes rapide orthogonale. Rappelons ici que cette transformation pyramidale orthogonale est principalement le résultat des travaux de S. Mallat et que son coût de calcul pour une image de taille $N \times M$ est de l'ordre de $(k \times N \times M)$ où k est la longueur de la séquence du filtre considéré. Cela sous-entend bien sûr que pour une image on utilise le même filtre selon les lignes et les colonnes, ce qui n'est pas toujours le cas dans les transformées en ondelettes. Par la suite nous ferons référence à cette transformation par l'abbréviation "OWT" pour Orthogonal Wavelet Transform, ce qui est plus précis que la dénomination courante de DWT ou "Discrete Wavelet Transform" qui ne devrait être employée que pour la version discrétisée (la DWT est un outil d'analyse numérique qui prolonge la représentation mathématique de cette transformée).
Revenons un moment à la forme littérale la plus simple de la transformation en ondelettes. Celle-ci, comme nous l'avons vu dans la première partie de ce mémoire, est un outil qui permet avant toute chose de représenter, pour chaque échantillon d'un ensemble de données, son degré de similitude avec une "petite onde" bien localisée dans l'ensemble de données. En particulier pour une image, la localisation correspond à la position de l'ondelette selon les lignes et les colonnes. Ajoutons que cette décomposition, en apparence analogue à la transformation de Fourier qui utilise une projection sur une base de fonctions continues *cosinus* et *sinus*, possède la particularité essentielle d'utiliser, en revanche, une base bien *localisée sur un support limité*, d'où son nom d'*ondelette*. Cette propriété de compacité du support présente d'ailleurs, si l'on considère de principe de Heisenberg, l'inconvénient de ne pouvoir être parfaitement définie à la fois dans le domaine de départ (l'espace pour les images) et dans le domaine dual (les coefficients d'ondelette ou les coefficients de Fourier). La similitude de chaque donnée avec l'ondelette est représentée par un *coefficient d'ondelette*. L'ensemble des coefficients d'ondelette constitue le *domaine transformé des ondelettes*. Enfin, nous avons parlé du positionnement de l'ondelette dans l'ensemble des données, qui correspond à la translation de la fenêtre dans la transformée de Fourier à fenêtre (STFT, Short Term Fourier Transform). Il nous faut encore mentionner, pour compléter la description de notre outil, le changement d'échelle, ou dilatation, de cette ondelette. Celui-ci nous permet, en refaisant la même opération de quantification de la similitude avec les données, pour plusieurs échelles de l'ondelette, de retrouver tous les éléments spectraux des données.

Regardons de plus près les propriétés du domaine des ondelettes. Celle-ci ont été résumées, classées et utilisées dans [RCB01, CB97, CNB98]. Nous utiliserons, dans notre modèle, certaines de ces propriétés, mais pas toutes. Nous ferons remarquer aussi, un peu plus loin, que certaines de ces propriétés "empiriques" ne conviendraient pas forcément dans tous les cas de figures. Nous nous sommes donc basés sur celles qui sont les plus fréquemment utilisées, à savoir les propriétés de type P1 et S2 (premier et deuxième groupe).
Voici quelles sont les propriétés *premières* des coefficients :

- P1 : Localisation. Chaque coefficient est une représentation *locale*, dans le domaine position-échelle, des données. Par représentation, nous rappelons qu'il s'agit d'un coefficient de similitude entre l'ondelette et les données (ou la fonction si l'on reste dans une représentation "continue", ou littérale ou encore mathématique).
- P2 : Multirésolution. Une analyse multirésolution permet la représentation "imbriquée" des sous-ensembles de coefficients d'ondelettes. Ceci n'est le cas QUE pour une multirésolution et l'on doit se référer aux conditions d'une multirésolution, [Mal01] par exemple, pour s'assurer de cette propriété. Celle-ci a d'ailleurs été aussi largement discutée dans la première partie de ce mémoire à propos de la possibilité d'utiliser l'algorithme "à trous" rapide dans une approche d'ondelettes adaptées au mouvement car, comme nous l'avons expliqué, cet algorithme nécessite que l'ondelette satisfasse à la condition de multirésolution (par notamment la *relation de double échelle*).
-P3 : Détection de contour. Les coefficients d'ondelette représentent naturellement ce que l'on appelle souvent les contours de l'image, ou en général de la donnée observée, et plus précisément l'ensemble des *singularités* de ces données.
Les propriétés P1 à P3 impliquent deux autres propriétés pour des images naturelles :
- P4 : Compacité de l'énergie. Hormis certains cas particuliers, dont les images fractales, la transformée en ondelette donne une *représentation creuse* des données initiales. En particulier pour une transformation orthogonale, les *coefficients d'ondelette*, qui sont des coefficients *passe-bande multi fréquence*, ne présentent une forte énergie qu'aux lieux des contours de l'image. Les autres coefficients sont de valeur très faible et en très grand nombre. Cette remarque permet déjà d'introduire la modélisation, adoptée plus loin pour les coefficients d'ondelette, par mélange de deux gaussiennes à moyenne nulle et à variances différentes : la première à forte variance pour les coefficients de haute énergie et l'autre à faible variance pour les coefficients de faible énergie.
-P5 : Décorrélation. Les coefficients d'ondelette d'images naturelles tendent à être approximativement décorrélés. Le terme approximativement provient de la constatation que les coefficients ont tendance à se regrouper localement. Cette propriété d'*essaimage* ne sera pas utilisée dans notre approche. En revanche, elle a fait l'objet de travaux sur la modélisation par *arbre de Markov caché* (HMT, Hidden Markov Tree) intra et inter-échelles [RCB01] des coefficients d'ondelettes et est utilisée pour la segmentation bayésienne. Le développement des méthodes HMT représente actuellement une part importante des travaux sur la modélisation statistique des images, sur les méthodes de débruitage, sur la séparation de source et sur la segmentation. Dans cette dernière application, d'autres modèles ont été développés comme le HMT-3S qui prend en compte la dépendance inter sous-bandes d'ondelettes pour chaque échelle.

Deux propriétés statistiques *secondaires* sont déduites du premier groupe de propriétés. Elles permettent de qualifier plus clairement la structuration des coefficients :
- S1 : Non-Gaussianité. Les propriétés P4 et P5 mettent clairement en évidence que les coefficients d'ondelette ne répondent pas à un modèle de gaussienne unique mais à une *loi de mélange de gaussiennes à deux composantes* (IGMM Independent Gaussian Mixture Model). Celle-ci est clairement établie par le fait que, pour chaque coefficient, on peut introduire une variable cachée z représentant l'état du coefficient : égal à 1 pour les faibles coefficients (W=weak) en fort nombre et égal à 2

pour les forts coefficients (S=strong) en faible nombre. Les coefficients d'état $z = 2$ voient aussi leurs valeurs changer considérablement dans toute l'image (forte variance). Ceci est dû au fait que les "contours" qu'ils représentent dans l'image ne possèdent pas tous la même force de singularité, ou coefficient de Hölder (ou encore de Lipshitz) à toutes les échelles. Ainsi les forts coefficients sont représentés par une distribution gaussienne centrée à forte variance, tandis que les faibles coefficients sont représentés par une gaussienne centrée à faible variance. On dira que la loi représentant les forts coefficients est très "piquée" car, bien que de forte variance, l'amplitude des coefficients est très élevée. Les figures 9.1 et 9.3 montrent le modèle de mélange à deux gaussiennes centrées dont la variance et l'amplitude sont caractéristiques des deux "classes" de coefficients d'ondelettes.

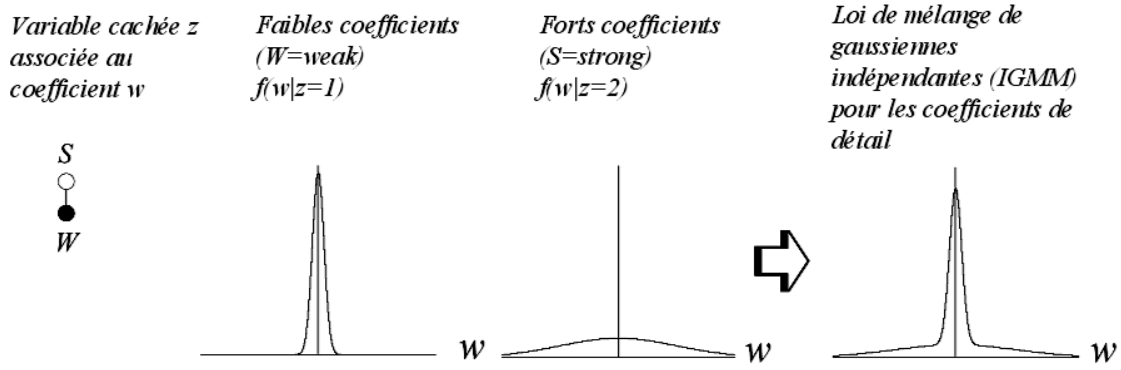


FIG. *9.1 – Modélisation des coefficients d'ondelette par une mélange de* 2 *gaussiennes (IGMM, Independent Gaussian Mixture Model) à moyenne nulle. A chaque coefficient peut être associé une variable de Markov cachée z représentant les deux états, faible (W=weak) ou fort (S=strong), que peut prendre un coefficient dans une sous-bande d'ondelette (détails). La variable z prend la valeur* $z = 1$ *dans le premier cas et* $z = 2$ *dans le deuxième. Les deux figures du centre montrent les lois conditionnelles* $f(w|z = 1)$ *et* $f(w|z = 2)$ *respectivement des coefficients faibles et forts. La loi* $f(w|z = 1)$ *présente un fort nombre de valeurs pour une faible variance. La loi* $f(w|z = 2)$ *présente les caractéristiques inverses. La figure de droite montre le mélange des deux lois (d'après [RCB01]). Les coefficients d'ondelette peuvent donc être segmentés en* $K = 2$ *classes.*

- S2 : Persistance. La valeur (forte ou faible) des coefficients d'ondelettes tend à se propager à travers les échelles d'une représentation en arbre quaternaire (quadtree) (voir Mallat et Zhong [MZ92, MH92] ainsi que Shapiro [Sha93]) . Cette propriété est aussi utilisée dans l'analyse fractale et en particulier dans les mesures de dimension mono et multifractale. Des fractales pures peuvent être des images synthétiques ou des images naturelles que l'on considère sous leur nature fractale après avoir filtré leur *caractère régulier*. Dans ce cas, on regarde l'évolution du coefficient d'ondelette à travers les échelles pour chaque site de l'image ou du signal. Le *regroupement* dont nous avons parlé ci-dessus ne consiste alors plus en la simple corrélation des amplitudes d'un coefficient

et de ses voisins entre deux étages (deux échelles) d'un arbre quaternaire, mais en la manière dont évolue l'amplitude de chaque coefficient à travers les échelles, représentant ainsi la force d'une singularité en tout point d'une image. Cet aparté sur l'analyse fractale se termine par les remarques suivantes : 1) cette méthode de "suivi" des forces des coefficients inter-échelles est appelée méthode des maxima de la transformée en ondelette (MMTO ou WTMM) car elle s'intéresse précisément à l'évolution des forts coefficients 2) c'est l'évolution de tous les coefficients d'une image qui permettra de déterminer soit sa dimension fractale, dans le cas d'une image de nature monofractale, soit son spectre $D(h)$ des singularités, où h est le coefficient de Hölder dont nous avons parlé plus haut, dans le cas d'une image multifractale.
Dans [CNB98, CB99], cette propriété de persistance permet d'obtenir, par apprentissage sur plusieurs images, la matrice de transition d'état des coefficients forts et faibles. Une estimation du maximum de vraisemblance de cette matrice, ainsi que des moyennes et des variances de mélanges, est réalisée par un algorithme d'Espérance-Maximisation (EM, Expectation-Maximisation). Ainsi est apparue la construction de modèles de Markov cachés dans le domaine des ondelettes, dans lesquels les états cachés ont une structure à dépendance Markovienne, avec des paramètres de mélange $\{\mu_{i,m}, \sigma^2_{i,m}\}$ et des probabilités de transition $P_{S_i|S_{\rho(i)}}(m|n)$.

D'autres propriétés *tertiaires* empiriques sont utilisées dans les modèles de type HMT et en particulier dans [RCB01]. Ces propriétés tertiaires ont essentiellement comme but de réduire le degré de complexité du modèle HMT. En effet le nombre élevé de paramètres à estimer ($4N^2$ pour une image carrée $N \times N$) en utilisant un modèle HMT prenant en compte toutes les propriétés précitées, conduit à un coût de calcul excessif. Ces propriétés tertiaires sont :
- T1 : Décroissance exponentielle inter-échelles.
- T2 : Persistance plus importante des coefficients aux hautes résolutions.

Ces dernières propriétés mériteraient d'être vérifiées dans différents cas. En particulier le caractère de décroissance exponentielle (T1) semble assez mal convenir à l'évolution inter-échelles de forts coefficients représentant des contours, d'après ce que nous avons dit sur l'analyse fractale et la MMTO. Ces propriétés T1 et T2 ont conduit les auteurs au développement d'un modèle "u-HMT" [RCB01], pour universal-HMT, permettant de réduire à 9 le nombre de paramètres à estimer.

Revenons sur la propriété P4 de compacité de l'énergie. Il a été démontré que dans les signaux et images réels, ou autres ensembles de données réelles, les coefficients d'ondelette de haute énergie, qui sont aussi les plus représentatifs dans l'image car ils représentent les contours (conjecture de Maar, voir aussi Mallat, Zhong et Hwang dans [MZ92, MH92]), présentent une nature creuse. Inversement, les coefficients faibles, dont l'importance est moindre car ils représentent des valeurs proches d'une même moyenne dans une région, sont en très grand nombre. C'est cette propriété des coefficients d'ondelettes qui lui a valu son nom de *représentation creuse* (sparse). Nous avons dit que grâce à cette propriété, la loi marginale des coefficients d'ondelette peut être représentée par le mélange d'une première gaussienne à forte variance (ou "longue queue") et d'une deuxième gaussienne indépendante à faible variance (Fig. 9.3).
Le modèle de mélange de deux gaussiennes indépendantes que nous adoptons pour les coefficients

d'ondelette devient particulièrement intéressant lorsqu'il s'agit d'appliquer notre méthode de segmentation bayésienne dans le domaine des ondelettes. Il signifie que la segmentation dans les sous-bandes d'ondelettes comporte seulement deux classes de régions, donc deux étiquettes : celle correspondant aux coefficients de forte amplitude et celle correspondant aux faibles amplitudes. L'approche que nous avons choisie consiste alors à effectuer la décomposition orthogonale de l'observable g sur J échelles, puis à réaliser la segmentation de la sous-bande d'approximation à la plus basse résolution avec un nombre élevé de classes. Ensuite la segmentation des sous-bandes successives de détail s'effectue avec seulement deux classes jusqu'à la résolution initiale de l'image. Ceci permet de réduire de façon très significative la complexité de la segmentation par le fait que 1) la segmentation avec un nombre élevé de régions ne se fait que sur une petite image qui est la sous-bande d'approximation à la résolution la plus faible, donc à une taille d'image 2^{N-J} et que 2) toutes les sous-bandes de détail sont segmentées avec deux étiquettes, ce qui accélère considérablement la détermination des paramètres des lois et l'algorithme de Gibbs dans chaque sous-bande de détail. Cette remarque faite, nous présentons dans ce qui suit la décomposition de notre observable dans le domaine des ondelettes orthogonales, puis nous décrirons l'algorithme complet de segmentation dans le domaine des ondelettes.

9.4 Projection dans le domaine des ondelettes

Afin d'appliquer l'approche bayésienne de segmentation au domaine des ondelettes, nous devons d'abord réaliser une projection de notre observable g par une *décomposition multirésolution orthogonale*. Nous utilisons ici la décomposition classique pyramidale de Mallat de complexité $mathcalONlog(N)$. Cette décomposition utilise des versions dilatées et translatées des fonctions d'échelle ϕ et ψ.
L'observable g peut alors s'écrire comme la somme des coefficients a_{J,b_1,b_2} et d^B_{j,b_1,b_2} de sa décomposition :

$$g(x,y) = \sum_{(b_1,b_2)\in\mathbb{Z}} a_{J,b_1,b_2}\phi^{LL}_{J,b_1,b_2}(x,y) + \sum_{B\in\mathcal{B}}\sum_{j\leqslant J}\sum_{(b_1,b_2)\in\mathbb{Z}} d^B_{j,b_1,b_2}\psi^B_{j,b_1,b_2}(x,y) \tag{9.1}$$

où $\phi^{LL}_{j,b_1,b_2} = 2^{-j}\phi(2^{-j}x-b_1, 2^{-j}y-b_2)$, $\psi^B_{j,b_1,b_2} = 2^{-j}\psi^B(2^{-j}x-b_1, 2^{-j}y-b_2)$ et $\mathcal{B} = \{HL, HH, LH\}$. Les sous-bandes HL, HH et LH sont appelées sous-bandes de détail, ou sous-bandes de coefficients d'ondelette. L et H représentent les filtres miroir passe-bande associés, aussi nommés couramment h et g. Donc HL correspond à la sous-bande de détails verticaux, HH à celle des détails diagonaux et LH à celle des détails horizontaux. LL est appelée sous-bande d'approximation, ou de coefficients d'échelle.

Les coefficients de la décomposition peuvent s'exprimer :

$$a_{J,b_1,b_2} = \int_{\mathbb{R}^2} g(x,y)\phi^{LL}_{J,b_1,b_2}dxdy \tag{9.2}$$

et

$$d^B_{j,b_1,b_2} = \int_{\mathbb{R}^2} g(x,y)\psi^B_{j,b_1,b_2}dxdy \tag{9.3}$$

Dans la suite nous utiliserons les notations V_J, W_j^V, W_j^D et W_j^H, respectivement pour les sous-bandes LL, HL, HH et LH. W_j^V, W_j^D et W_j^H sont respectivement les sous-bandes de détails verticale, diagonale et horizontale. Les filtres d'ondelette correspondant sont donnés par :

$$\begin{cases} W_j^V & \rightarrow & \psi_j \phi_j[\vec{b}] \\ W_j^D & \rightarrow & \psi_j \psi_j[\vec{b}] \\ W_j^H & \rightarrow & \phi_j \psi_j[\vec{b}] \end{cases} \tag{9.4}$$

La décomposition de notre observable g est réalisée à partir de l'image à sa résolution initiale $L = 0$, ou encore 2^L, jusqu'à l'échelle J, ou 2^J. Décomposer sur cinq échelles signifie que $J = 5$ et que le paramètre d'échelle prend les valeurs $j \in 1, 2, ..., J = 5$. A la reconstruction, nous sommons tous les coefficients de la décomposition comme indiqué dans la relation 9.1. La résolution correspondant à une échelle est obtenue par la puissance inverse 2^{-j}. La confusion entre échelle et résolution est fréquente et par la suite nous tentons de rester aussi clairs que possible dans la description de l'algorithme à partir de la plus haute échelle (basse résolution).

En ce qui concerne le choix de l'ondelette, et puisque nous nous intéressons essentiellement aux sous-bandes d'ondelette, hormis la première sous-bande d'approximation, une ondelette adaptée aux fortes discontinuités se présentait comme une bonne candidate. C'est donc l'ondelette de Haar, ou plus simplement la "fonction de Haar", que nous avons finalement adoptée et qui est utilisée dans l'algorithme que nous décrivons plus loin en détail. Nous avons aussi évalué les ondelettes de Daubechies, les symlettes et les Coiflettes (Coifman). Celles-ci présentent l'avantage d'être adaptées à une multirésolution, de posséder une fonction d'échelle (donc d'être orthogonales) et d'être à support compact. Cette dernière propriété, sur laquelle nous avons insisté dans la première partie, n'est en fait que très relative. L'ondelette ne peut être que "relativement" compacte à la fois dans l'espace de départ et dans l'espace dual de Fourier. Ceci toujours en raison du principe de Heisenberg qui nous rappelle que l'on ne peut être simultanément bien défini dans l'espace et le temps (ou les fréquences) que dans une certaine "mesure". Cette mesure est donnée par le nombre $1/4\pi$ qui précise que des boîtes, contenant l'ondelette, et de taille $\Delta\sigma_t \times \Delta\sigma_\nu$, où $\Delta\sigma_t$ et $\Delta\sigma_\nu$ sont respectivement l'écart-type en temps et l'écart-type en fréquence de l'ondelette, ne peuvent avoir une taille inférieure à $h/4pi$. Ces ondelettes présentent aussi une faible régularité, ce qui est justement intéressant dans notre approche. Cependant les résultats obtenus avec l'ondelette de Haar se sont avérés les meilleurs. Nous rappelons que cette "ondelette" est obtenue par la multirésolution de fonctions constantes par morceaux. Elle possède le support le plus compact parmi toutes les ondelettes orthogonales et possède un seul moment nul ce qui en fait la moins régulière des fonctions d'échelle "lissantes".
Les figures ci-dessous décrivent l'application de la transformée orthogonale sur une bande spectrale d'une image hyperspectrale (contenant 224 bandes spectrales) d'une vue satellite. Les figures 9.3 b) et c) montrent respectivement les histogrammes de la sous-bande basse résolution d'approximation et de la sous-bande basse résolution diagonale de détails. Comme nous l'avons dit précédemment, le premier histogramme peut être modélisé par un mélange des plusieurs gaussiennes indépendantes, ce qui motive la segmentation de cette sous-bande d'approximation par un nombre relativement élevé d'étiquettes (couramment 4 à 10). Inversement, le deuxième histogramme de la sous-bande

diagonale de détail peut être modélisé par un mélange de deux gaussiennes indépendantes (qui est plus visible dans la représentation lin-log), l'une de forte variance (coefficients de fortes amplitude et importance en faible nombre) et la deuxième de faible variance (nombreux coefficients de faible valeur). Le résultat est, nous l'avons dit, que la segmentation dans toutes les sous-bandes de détail peut être réalisé avec seulement deux étiquettes $K = 1$ ou $K = 2$.

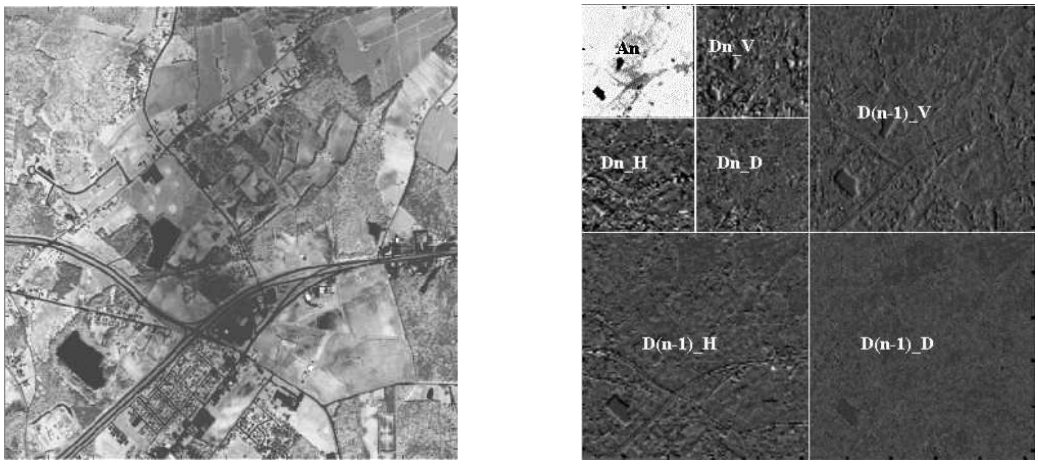


FIG. *9.2 – a) Image originale de la bande 100 d'une image satellite hyperspectrale composée de 224 fréquences. b) Représentation pyramidale de Mallat de la décomposition orthogonale rapide avec deux échelles appliquée au canal 100 de notre image hyperspectrale. Ici la lettre n peut être remplacée par J, le paramètre d'échelle* 2^J *correspondant à la plus basse résolution.*

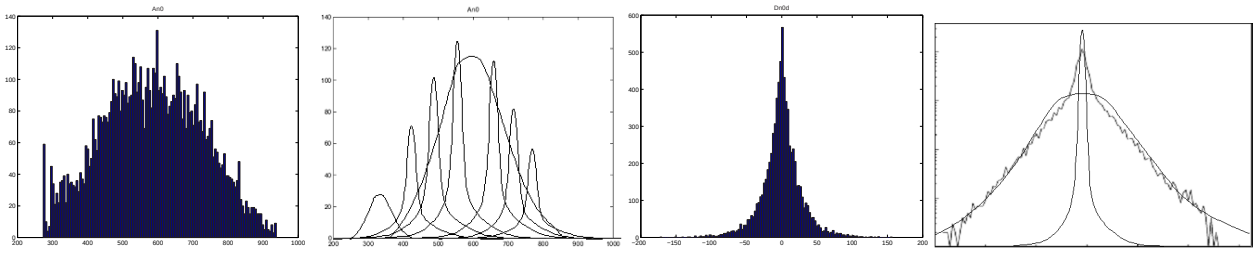


FIG. *9.3 – a) Histogramme des* A_J *(ou* A_n*), les coefficients d'approximation à la plus basse résolution. b) L'histogramme des coefficients* A_J *à la plus basse approximation est décrit par un mélange de gaussiennes, chacune représentant une classe c) Histogramme des* D_J^D *(or* D_n^D*), les coefficients diagonaux de détail à la plus basse résolution. d) Histogramme Linlog des* D_J^D *expliquant le choix du modèle de mélange de deux gaussiennes indépendantes, l'une à large variance, l'autre à faible variance.*

9.5 Développement du modèle de Potts pour la transformée orthogonale rapide

Afin de trouver des régions statistiquement homogènes pour les segments, notre méthode de segmentation bayésienne dans le domaine direct utilise un champ aléatoire [FMD05] de Potts-Markov pour définir la dépendance spatiale des étiquettes dans le HMM. Dans ce modèle direct, la dépendance des étiquettes des pixels est recherchée dans un voisinage d'ordre 1. Comme la segmentation est faite dorénavant dans le domaine des ondelettes, il est intéressant de tenir compte du fait que cette dépendance spatiale suit des directions privilégiées par les orientations des trois sous-bandes de "détail" qui sont les directions verticale, diagonales ($D_1 = \pi/4 \quad$ et $\quad D_2 = 3\pi/4$) et horizontale. Cette connaissance a priori est prise en considération en introduisant ces orientations dans un nouveau modèle du champ aléatoire de Potts-Markov qui prend en compte les voisinages d'ordre 1 et 2 (voir Fig. 9.4). Le nouveau modèle du PMRF s'écrit alors :

$$p(z(i,j),(i,j)\in\mathcal{R}) = \frac{1}{T(\alpha_V,\alpha_{D_1},\alpha_{D_2},\alpha_H)} \times \exp\Bigg\{ \begin{array}{l} +\alpha_V \sum\limits_{(i,j)\in\mathcal{R}} \delta(z(i,j)-z(i-1,j)) \\ +\alpha_{D_1} \sum\limits_{(i,j)\in\mathcal{R}} \delta(z(i,j)-z(i+1,j-1)) \\ +\alpha_{D_2} \sum\limits_{(i,j)\in\mathcal{R}} \delta(z(i,j)-z(i-1,j-1)) \\ +\alpha_H \sum\limits_{(i,j)\in\mathcal{R}} \delta(z(i,j)-z(i,j-1)) \Bigg\} \end{array} \tag{9.5}$$

où (i,j) sont les coordonnées de chaque site de l'image.

Les paramètres α_V, α_{D1}, α_{D_2} et α_H contrôlent respectivement le degré de dépendance spatiale de la variable z dans les directions V, D_1, D_2 et H. Pour la sous-bande diagonale de d'ondelettes, nous rassemblons les dépendances D_1 et D_2 en une seule dépendance diagonale D_1D_2. Pour la première sous-bande d'approximation de basse résolution, nous rassemblons également les dépendances V et H en une dépendance VH qui est équivalente à la dépendance du premier ordre (4-connexité) utilisée pour la segmentation présentée initialement dans le domaine direct [FMD05].
Afin de réaliser une mise en oeuvre rapide de l'algorithme MCMC de Gibbs dans le domaine direct, nous avons montré dans la section 7.6 qu'il est possible d'accélérer l'échantillonnage pour tous les sites de l'image en effectuant cette mise en oeuvre "en parallèle" [FMD05]. De la même façon pour les dépendances diagonales D1 et D2, nous pouvons aussi réaliser une mise en oeuvre en parallèle de l'échantillonneur de Gibbs. Dans ce cas les deux ensembles de sites indépendants ne sont plus disposés en échiquier mais l'image est partagée en deux sous-ensembles de sites de lignes noires et blanches entrelacées, de la même façon que l'alternance des lignes dans un *balayage vidéo entrelacé* (Fig. 9.4). Ceci permet de considérer tous les sites d'un même sous-ensemble (blancs ou noirs) comme indépendants conditionnellement à la connaissance des sites de l'autre sous-ensemble (respectivement noirs ou blancs). Ainsi, une itération complète de l'échantillonneur de Gibbs, pour les sous-bandes diagonales, sera réalisée en recherchant les valeurs des sites voisins

diagonaux à partir de tous les "sites blancs" puis à partir de tous les "sites noirs". L'application de ce modèle aux sous-bandes d'ondelettes est faite en supprimant les termes du PMRF qui ne s'appliquent pas à la sous-bande concernée. Ce nouveau modèle prend en compte le voisinage d'ordre 2 au lieu du voisinage d'ordre 1. De la même façon que pour la sous-bande d'approximation en dépendance "VH", les paramètres α_{D_1} et α_{D_2} utilisés dans la relation 9.5 peuvent être ajustés à une valeur positive variable ce qui permet d'attribuer un paramètre de dépendance variable entre les étiquettes des régions concernées. Comme nous l'avons vu dans la section 7.4, ceci signifie que plus le paramètre α est élevé, plus l'a priori d'avoir un faible nombre de régions de grande taille est élevé. Nous avons utilisé cette possibilité de régler indépendamment les dépendances dans la sous-bande d'approximation et dans les sous-bandes de détail. Ceci nous a permis de constater un regroupement en régions plus larges selon les orientations verticale, diagonale et horizontale. Nous pensons que le réglage indépendant des paramètres α des différentes sous-bandes pourrait apporter une information supplémentaire pour la segmentation en présence de zones fortement texturées.

9.6 Segmentation bayésienne dans le domaine des ondelettes

9.6.1 Description de l'algorithme

Notre schéma de segmentation dans le domaine des ondelettes est basé sur la segmentation itérative de toutes les sous-bandes depuis la bande d'approximation de plus basse fréquence en remontant jusqu'à la résolution de l'image (Fig. 9.5). Comme nous l'avons dit, la segmentation bayésienne est effectuée sur les coefficients des sous-bandes d'ondelette, et à toutes les échelles, en n'utilisant que deux classes. En ce qui concerne la première sous-bande d'approximation, la valeur K_A du nombre de classes est choisie en fonction du nombre de classes que l'on désire obtenir dans la segmentation finale. Comme nous l'avons vu dans la description du modèle direct, cette valeur K_A peut éventuellement être choisie supérieure au nombre réel de classes de l'image (image synthétique par exemple). Dans ce cas la valeur K_A sera automatiquement réduite à la valeur K_e, ou nombre de classes effectives, de l'image. L'algorithme de segmentation peut être décrit par les étapes suivantes :

1) Décomposition en ondelettes à l'ordre J, c'est-à-dire jusqu'à la plus basse échelle 2^J, sur une base orthogonale (ondelette de Haar).

2) Segmentation des coefficients d'approximation V_j à l'échelle 2^J avec le nombre de classes désirées dans la segmentation finale, par exemple $K = 8$. Afin que l'algorithme de Gibbs converge vers une segmentation stable pour des valeurs élevées de K, et/ou pour des images de complexité importante (nombre de régions très élevé), on attribue de préférence une valeur élevée au nombre d'itérations.

3) Dans l'image segmentée des coefficients d'approximation $z(v_j)$, nous détectons (par dérivation) les régions présentant des discontinuités verticales, diagonales ($\pi/4$ et $3\pi/4$) et horizontales.

4) A cette même échelle, les 3 sous-bandes de détail W_j^v, W_j^d et W_j^h sont segmentées et nous employons respectivement, comme initialisation de ces segmentations, les 3 sous-ensembles de discontinuités $diff_v$, $diff_d$ et $diff_h$ calculés à l'étape précédente. Cette étape est réalisée avec un nombre faible de classes ($K = 2$) et d'itérations de l'échantillonneur de Gibbs.

5) Les 3 sous-bandes de détail segmentées sont suréchantillonnées d'un facteur 2 afin de répéter le

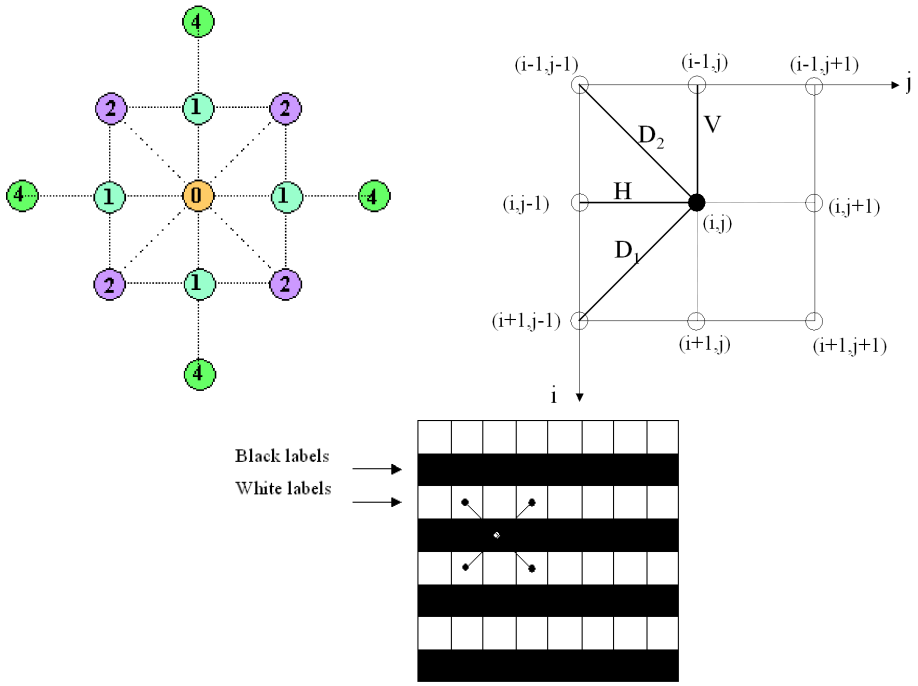


FIG. *9.4 – a) Champs de Markov d'ordre 1, 2 et 4 (d'après [Idi01]). Les modèles de Markov d'ordre 1 et 2, utilisés pour la dépendance des étiquettes des pixels, correspondent à un voisinage de type "8-connexité". b) Dépendances d'un site (i, j) avec son voisinage d'ordre 1 (directions V, H) et avec son voisinage d'ordre 2 (directions D_1, D_2), utilisés dans notre nouveau modèle de PMRF. c) Mise en parallèle de l'échantillonnage des sites pour les sous-bandes diagonales. La mise en parallèle est faite en échantillonnant simultanément tous les sites des lignes noires, qui peuvent être considérés comme indépendants conditionnellement à la connaissance des sites des lignes blanches, puis tous les sites des lignes blanches. Un échantillonnage de Gibbs sur toute l'image est ainsi, de la même façon que pour le modèle du premier ordre, réalisé en deux coups d'échantillonnage successifs sur chaque demi-image.*

processus à l'échelle immédiatement inférieure.

6) Nous segmentons les 3 sous-bandes de détail $W^v_{(j-1)}$, $W^d_{(j-1)}$ et $W^h_{(j-1)}$ en utilisant comme initialisation de la segmentation les résultats de l'étape précédente. Le même processus est ainsi répété jusqu'à la résolution de l'image.

7) Nous reconstruisons l'image segmentée à partir de la plus basse résolution (échelle 2^J) de la décomposition. La reconstruction utilise :
- pour la sous-bande initiale d'approximation (échelle 2^J) : la moyenne des coefficients originaux

d'échelle dans chaque région de la sous-bande segmentée : $(\bar{v_j}(z_k))$ avec $k \in 1...\ de, k$.

- pour toutes les sous-bandes de détail : les coefficients originaux d'ondelette, $W_{j\in\{1...j\}}^{(v,d,h)}$, pour les segments qui appartiennent à la classe $k = 2$. Les coefficients appartenant à la classe $k = 1$ sont annulés.

8) Nous reclassifions l'histogramme, dans le domaine direct, de l'image segmentée obtenue à l'étape précédente. Ceci est réalisé en recherchant les seuils, $j \in \{1...(k+1)\}$ des modes de l'histogramme de l'image reconstruite dans le domaine direct. Cette reclassification est nécessaire parce qu'il n'y a aucune raison pour que les coefficients d'ondelette appartiennent, après inversion, à la classification en K classes réalisée dans la sous-bande d'approximation. Cette étape est facilement réalisée en employant les K valeurs des moyennes m_k calculées dans la sous-bande d'approximation V_J. Les seuils des modes de l'histogramme brut de l'image reconstruite sont obtenus en prenant simplement le point milieu des moyennes pour chaque couple successif de classes : $Th_M^j = \frac{m_{k+1}+m_k}{2}$. Nous prenons également comme valeurs du premier et du dernier seuil, les valeurs minimale et maximale des pixels de l'image.

9) La reclassification de l'histogramme est faite dans la même boucle que le nouvel étiquettage des pixels en K classes successives, ce qui fournit en même temps l'image finale segmentée en K classes avec comme première valeur $K = 1$.

D'un premier point de vue, cette approche, basée sur l'initialisation de la segmentation des sous-bandes de détail, à la plus basse échelle, à partir des discontinuités de la sous-bande d'approximation, peut être considérée comme une façon de prendre en compte la dépendance *intra-échelle* des coefficients d'ondelette. D'un second point de vue, l'initialisation de la segmentation des sous-bandes de détail à une échelle j, par la segmentation des sous-bandes de détail à l'échelle immédiatement supérieure $j + 1$, peut également être considérée comme un moyen de prendre en compte la dépendance *inter-échelles* des coefficients d'ondelette.

Remarque : Les seuls paramètres à choisir avant de commencer le calcul sont le nombre maximum d'itérations, $itermax$, de l'algorithme d'échantillonnage de Gibbs, et le nombre K de classes demandées. $Itermax$ est généralement pris assez grand pour assurer la convergence de la segmentation, c'est-à-dire entre 20 et 50. K doit être pris au moins égal au nombre de classes dans l'image, si cette valeur est connue. La valeur K de la volonté diminue automatiquement pour s'adapter au nombre maximum des classes dans l'image, comme nous l'avons déjà dit, mais n'augmentera pas. Si le nombre de classes est inconnu, alors K doit être choisi en fonction de l'application. Dans une image naturelle, ce nombre est généralement pris entre, environ, deux et dix. Parmi les autres paramètres réglables nous avons aussi la valeur de α_{potts}, c'est-à-dire $\alpha_V, \alpha_W^V, \alpha_W^D$ et α_W^H pour la sous-bande d'approximation et les sous-bandes de détail. Ces paramètres sont généralement fixés à 1 mais peuvent être néanmoins ajustés individuellement. L'augmentation de α_V permet d'obtenir des zones homogènes plus grandes dans la sous-bande d'approximation. En ce qui concerne les sous-bandes de détail, nous avons fait plusieurs essais avec des valeurs de α_W^V, α_W^D et α_W^H variant entre

$0,5$ et 5. Nous avons constaté que dans des images présentant une texture, les coefficients forment des zones plus homogènes selon la direction des composantes de la texture. Il est possible que l'utilisation individuelle du paramètre de Potts dans les sous-bandes de détail, et pour un modèle mieux adapté à des images texturées, en facilite la segmentation. Nous avons constaté dans les tests réalisés qu'une trop grande variation par rapport à la valeur $\alpha_{potts} = 1$ ne donnait pas de meilleurs résultats. Un autre paramètre initial important est z_0, la segmentation initiale de l'observable, si nous en possédons une connaissance a priori. C'est ce même paramètre d'initialisation que nous employons dans notre algorithme, entre les sous-bandes, pour diminuer le temps de segmentation. Les paramètres m_k, v_k et v_ϵ peuvent également être fixés au départ si nous les connaissons. Dans tous les autres cas, on attribue à l'ensemble de ces paramètres une valeur initiale par défaut et leur calcul est réalisé automatiquement suivant les loi a priori, a posteriori et lors de l'échantillonnage.

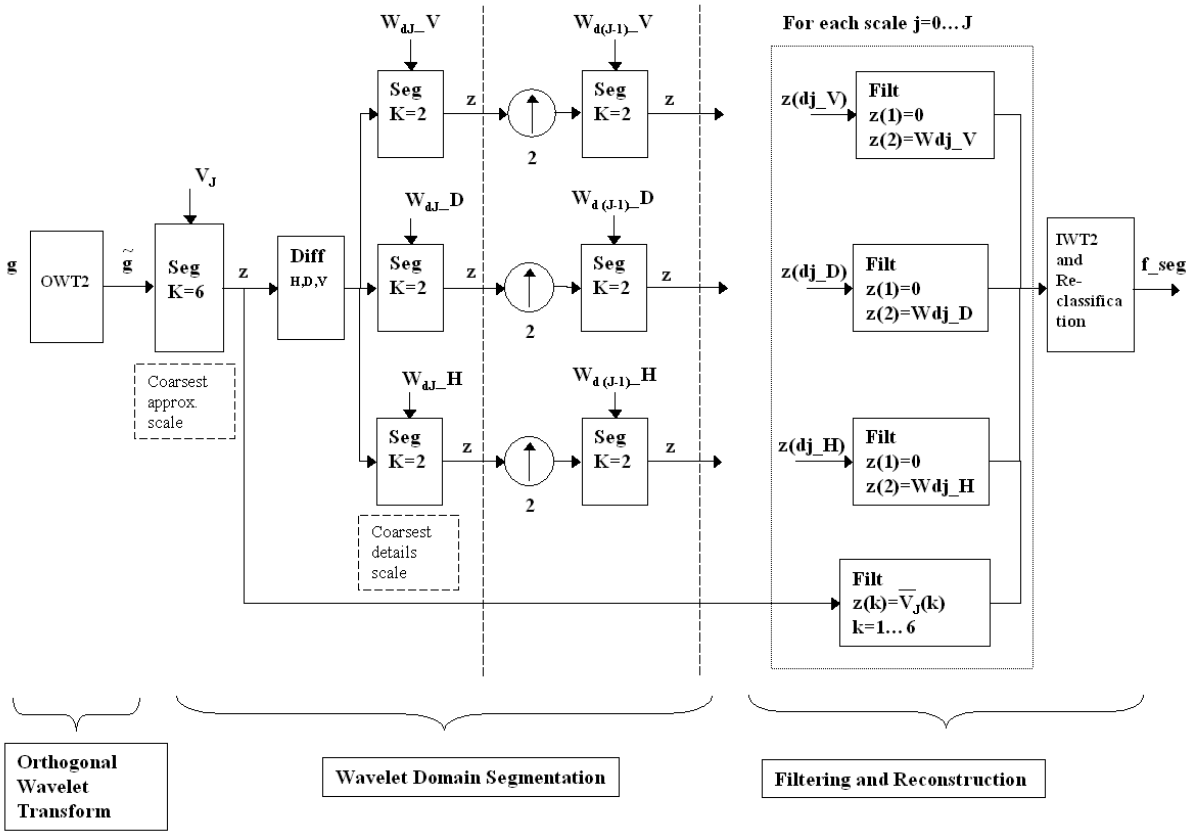


FIG. *9.5 – Schéma de segmentation bayésienne dans le domaine ondelettes. La donnée observée g est décomposée dans le domaine ondelettes (deux niveaux de décomposition dans cet exemple), segmentée dans ce domaine avec* 6 *classes pour la sous-bande d'approximation et deux pour les sous-bandes de détail. La sous-bande d'approximation est filtrée en remplaçant la valeur* $z(r) = k$ *de chaque classe* k *par la moyenne des coefficients d'échelle originaux dans cette région* $z(r)$*. Toutes les sous-bandes d'ondelettes sont filtrées en mettant à zéro les coefficients appartenant à la classe* $k = 1$ *des coefficients "faibles" et en laissant les coefficients de classe* $k = 2$ *à leur valeur initiale. La segmentation finale est obtenue par reconstruction dans le domaine direct et par reclassification de l'histogramme.*

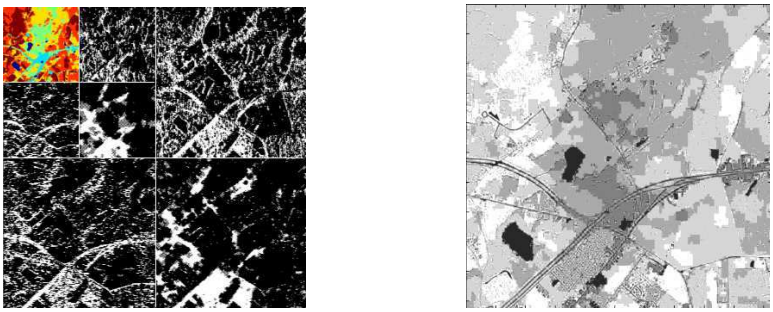


FIG. *9.6 – a) Représentation pyramidale de Mallat de l'image segmentée z. La bande d'approximation est segmentée en 6 classes (imagette en couleurs). Les sous-bandes de détail sont représentées en noir pour la classe des coefficients faibles, qui sont mis à zéro, et en blanc pour la classe des coefficients forts qui seront utilisés pour la reconstruction finale. b) Résultat de la segmentation après reconstruction dans le domaine direct.*

Chapitre 10

Résultats et comparaisons

Remarque : Lors de nos premiers essais nous avons comparé les résultats de la segmentation dans le domaine ondelettes avec ceux du domaine direct. Dans le domaine ondelettes, nous avons testé la segmentation avec des ordres de décomposition plus ou moins élevés. Nous savons que la reconstruction de l'image, à partir de la sous-bande basse d'approximation et de toutes les sous-bandes d'ondelettes, donnera exactement l'image initiale. De la même façon si nous éliminons les coefficients d'ondelette de classe $K = 1$, donc tous les coefficients de faible importance, cela revient à une méthode de compression, voire de débruitage, si l'on fixe un seuil ("dur" ou "mou") aux coefficients, et la reconstruction se fera avec une faible perte. Par ailleurs, à la limite d'une décomposition qui descendrait à la taille d'un pixel pour la sous-bande la plus basse, on sait que la reconstruction sera toujours complète. Dans ce cas limite, la sous bande d'approximation ne fournit d'ailleurs qu'une seule valeur qui correspond à *la valeur moyenne* sur toute l'image, et la reconstruction peut se faire, au décalage près de cette valeur moyenne, avec seulement toutes les sous-bandes d'ondelettes. En revanche, à cette même limite, il est évident que la segmentation de la sous-bande d'approximation n'a plus aucune signification. En pratique, pour rendre significative la segmentation de l'image à partir de la sous-bande d'approximation de l'échelle basse, il faut limiter le nombre de niveaux de la décomposition de façon à obtenir pour la sous-bande de basse résolution un minimum de 32×32 pixels . Le choix de l'ordre 2 ou 3 est donc un choix raisonnable pour des images de 512×512. La décomposition sur un nombre de niveaux plus importants montre que la reconstruction fait apparaître des artefacts dus à la "pixellisation" de l'image, autrement dit au fait que les discontinuités et les coefficients de détail auxquels nous nous intéressons aux basses résolutions, vont présenter une forme en "marches d'escalier" qui correspond à la forme carrée des pixels. Pour une image de 1024×1024 il est possible de faire une décomposition sur quatre niveaux sans risquer de supprimer les régions de faible dimension. Nous montrons d'ailleurs un peu plus loin, dans les résultats de la segmentation de la mosaïque synthétique de la base SIPI [SIP], qu'un ordre de décomposition de quatre va donner apporte un très bon résultat.

Les résultats comparatifs de cette section ont été effectués sur trois types d'images test. La première image test est une image synthétique de mosaïque, "texmos3.s1024" que nous avons suréchantillonnée

à partir de l'image texmos3.512 de la base de données SIPI ([SIP]) et sur laquelle nous avons ajouté des bruits gaussiens de différente variance pour les régions et pour l'image totale.
La deuxième image test est une image naturelle, extraite aussi de la base de données SIPI, qui représente la côte de San Diego à la "pointe Loma".
La troisième image correspond à l'une des 224 bandes spectrales contiguës d'une image satellite hyperspectrale AVIRIS "$3D$".
Quatre algorithmes de segmentation sont utilisés pour la comparaison : (1) la segmentation bayésienne de Potts-Markov dans le domaine direct (BPMS), (2) la segmentation bayésienne dans le domaine des ondelettes (WBPMS) développée dans ce chapitre, (3) la segmentation HMT présentée dans [CB01b] et (4) deux versions de l'algorithme de regroupement "K-MEANS" utilisé avec deux mesures de distance différentes.
Pour le premier exemple nous avons testé les algorithmes 1, 2 et 4. Pour le deuxième exemple nous avons testé les algorithmes 1, 2 et 3. Pour le dernier exemple (notre image hyperspectrale), nous avons comparé les méthodes 1 et 2.
Afin d'évaluer la qualité de segmentation, nous utilisons le critère numérique "Pa" qui donne le pourcentage de précision (Percentage of accuracy) de la classification, c'est-à-dire le pourcentage de pixels correctement classifiés [SF03b, SF03a]. Ce test est employé évidemment pour des images synthétiques (notre premier essai) avec une classification z connue.
Nous indiquons également, pour chacun des essais, le temps de calcul qui est un enjeu important. Tous les calculs ont été faits sur un Pentium M 1.6Mhz, avec 1MB de RAM-cache et 512MB de RAM-système, qui n'est donc pas une machine particulièrement dédiée au test d'algorithmes de traitement d'image et encore moins d'algorithmes parallélisables et coûteux en calcul comme l'échantillonneur de Gibbs. Ceci nous laisse supposer que sur une machine dédiée, nous pourrions atteindre des temps de calcul nettement inférieurs.
Pour l'ensemble des images nous affichons la plupart du temps l'image originale et son histogramme, puis la segmentation finale et son histogramme et montrons les quelques étapes intermédiaires importantes. Ce sont notamment la segmentation de la sous-bande d'approximation et son histogramme ainsi que le résultat de la segmentation après reconstruction dans le domaine direct et avant reclassification par la méthode du seuillage de l'histogramme.

10.1 Premier exemple

Dans cet exemple nous utilisons la mosaïque "texmos3.s1024" de la base de données SIPI (Signal and Image Processing Institute) de l'USC (University of Southern California). Cette mosaïque est une image 512×512, 8 bits, suréchantillonnnée à 1024×1024 et composée de 23 régions homogènes (non texturées) et 8 classes. Nous avons suréchantillonné cette image afin, comme nous l'avons dit dans la remarque ci-dessus, montrer les capacités de l'algorithme de segmentation WBPMS sur quatre niveaux de décomposition. La segmentation démarre donc sur une sous-bande d'approximation de taille 64×64. C'est sur cette image de test que nous avons ajouté un bruit gaussien de variance différente pour chaque classe additionné d'un bruit global gaussien avec une autre valeur de variance.
La figure 10.1 montre l'image originale $\boldsymbol{f}$, l'image observée $\boldsymbol{g}$ et l'histogramme de l'image $\boldsymbol{g}$.

10.1.1 Image de test mosaïque texmos3.s1024

Nous n'avons pas représenté ici l'histogramme original de cette image synthétique car il est exactement composé des huit classes de cette image. Nous montrons l'image modifiée par des bruits gaussiens de différentes variances et son histogramme, où les modes entre les trois dernières classes deviennent difficilement détectables.

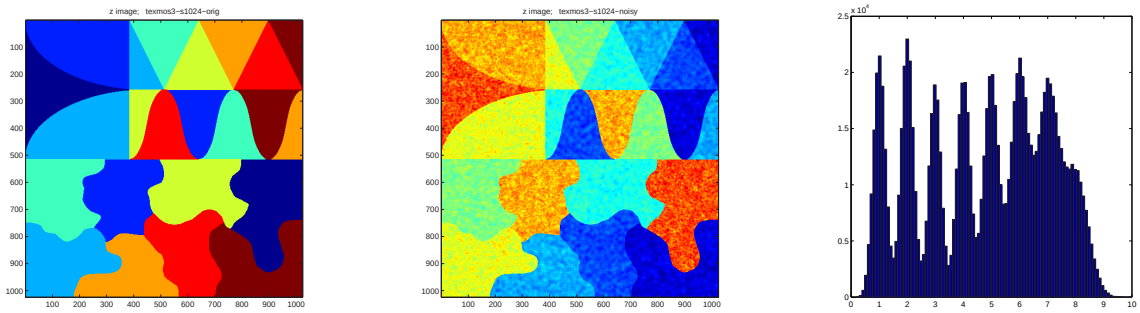


FIG. *10.1 – a) Image f originale de la mosaïque texmos3s de la base "SIPI", suréchantillonnée à* 1024×1024 *et montrant les* 23 *régions partagées en* $K = 8$ *classes. b) mosaïque texmos3s perturbée par des bruits additifs gaussiens de variance différente pour chaque classe et additionnés d'un bruit global gaussien avec une autre valeur de variance. c) Histogramme de la mosaïque bruitée.*

10.1.2 Résultat avec la méthode BPMS : Segmentation par PMRF dans le domaine direct

La méthode BPMS a été testée avec 20 itérations 10.2. Le nombre de classes demandées et obtenu est 8. Le temps de calcul est de 1805 secondes avec un pourcentage de précision $P_a = 80.34\%$. Nous savons par expérience que cette méthode requiert un nombre beaucoup plus élevé d'itérations pour atteindre une convergence de l'algorithme vers une bonne segmentation. Ce nombre est généralement compris entre 100 et 1000. Cependant pour permettre une comparaison significative avec notre méthode dans le domaine des ondelettes, nous avons choisi d'affecter le même nombre, faible, d'itérations aux deux méthodes BPMS, WBPMS. La méthode par regroupement (K-means) ne nécessite aucune initialisation du nombre d'itérations. Nous savons en revanche que le nombre de 20 itérations est suffisant à la méthode WBPMS pour atteindre la convergence, ce qui est montré dans la section suivante.

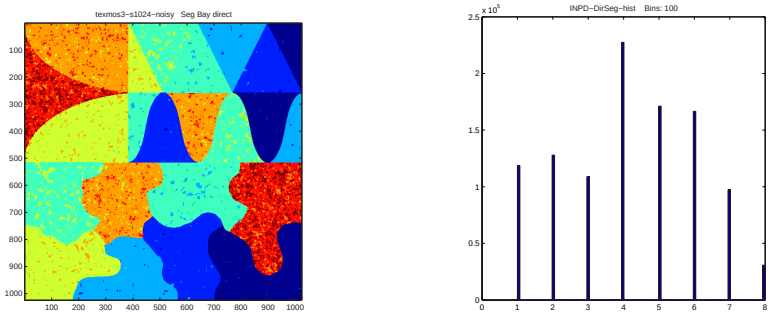


FIG. *10.2 – a) Résultat de la segmentation bayésienne dans le domaine direct avec les paramètres* $K = 8$ *et* $itermax = 20$. *Le pourcentage de bonne classification* $Pa = 80.34\%$. *Le temps de calcul est 1805s b) Histogramme final en huit classes.*

10.1.3 Résultat avec la méthode WBPMS : Segmentation par PMRF dans le domaine ondelettes

La méthode WBPMS a été testée avec 2 valeurs d'échelle maximale de décomposition : $J = 4$ et $J = 3$. Pour les deux valeurs, le nombre d'itérations maximum est de 20 à la fois pour les sous-bandes de coefficients d'échelle et d'ondelettes. Avec $J = 4$, nous avons obtenu un taux de bonne classification $Pa = 94.8\%$ dans un temps $t = 260s$. Avec $J = 3$, nous avons obtenu $Pa = 98.06\%$ dans un temps de $t = 384s$. Ces deux résultats montrent un très comportement quantitatif, sur une image de test, de notre méthode WBPMS. Nous montrons dans les figures suivantes les résultats de segmentation après segmentation de la sous-bande d'approximation et avant la reclassification dans le domaine direct. Comme nous l'avons dit, la méthode peut se heurter à deux écueils. Le premier est de choisir un nombre de décomposition trop élevé ce qui résulte alors en une disparition de régions de l'image et à une "sous-classification" de l'image ; par exemple le risque est de ne détecter que 7 classes dans une image qui en comporte exactement 8. Les figures ci-dessous montrent que cette règle du nombre limité de niveaux de décomposition a été respectée. Le deuxième écueil est, dans une image comportant un nombre important de détails et des zones homogènes trop petites, de ne pas détecter, dans l'histogramme avant reclassification finale, un nombre correct de classes "prédominantes". Les figures ci-dessous montrent aussi que dans les deux cas $J = 4$ et $J = 3$, pour cette image test, les histogrammes avant reclassification comportent bien les 8 classes théoriques et que la reclassification ne filtre qu'un petit nombre de valeurs. Si le nombre de coefficients, issus des sous-bandes de détail, donnait trop de valeurs entre les classes "dominantes", ces classes pourraient être noyées parmi les valeurs en question et la reclassification finale montrerait un nombre de classes faux.

• Résultats avec une décomposition sur 4 niveaux.

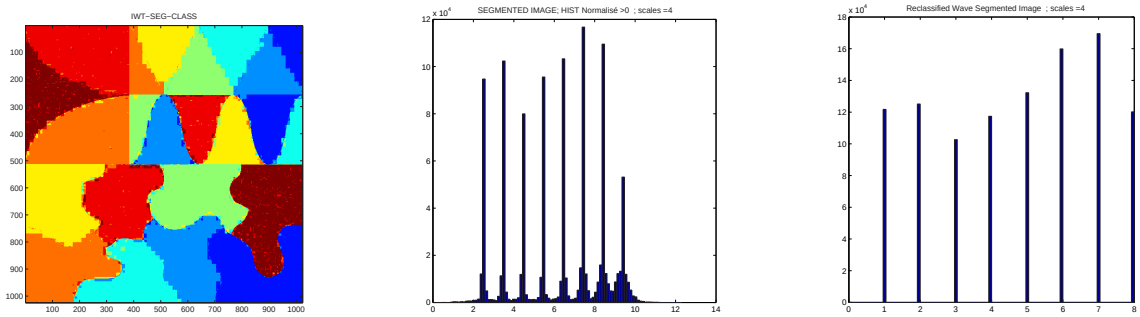


FIG. *10.3 – a) Segmentation dans la sous-bande d'approximation à la résolution la plus faible (échelle $j = 2 = J$) suivie du moyennage des coefficients dans chacune régions. b) Segmentation finale de toutes les sous-bandes présentée sous la forme pyramidale de Mallat. Nous pouvons constater que les coefficients d'approximation sont segmentés avec les* 8 *des classes demandées (plus visible sur l'histogramme c.), et que les sous-bandes de détail sont segmentées en* 2 *classes. c) Histogramme "brut" de la segmentation finale après reconstruction et sans reclassification. Cet histogramme montre que les valeurs reconstruites peuvent être négatives et qu'elles ne suivent pas rigoureusement* 8 *classes distinctes. Les coefficients de détail reconstruits sont répartis autour des trois dernières classes.*

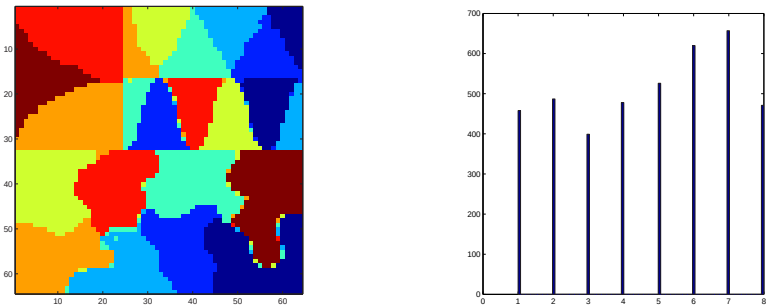


FIG. *10.4 – Following of Fig. 10.3. a) Segmentation de la sous-bande V_4 b) Histogramme de la sous-bande V_4 montrant que les* 8 *classes ont bien été déjà détectées lors de la segmentation de la sous-bande d'approximation.*

• Résultats avec une décomposition sur 3 niveaux.

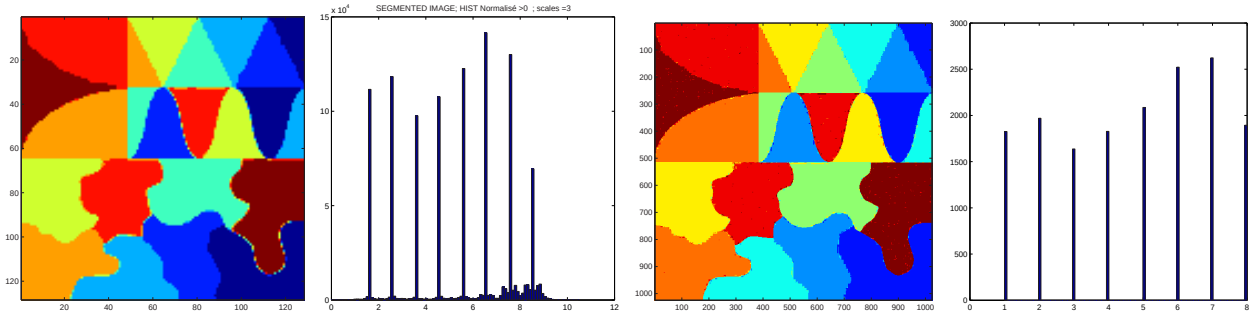


FIG. *10.5 – Résultat de la segmentation avec* $J = 3$ *et itermax* $= 20$ *pour les deux types de sous-bandes. Ici nous atteignons une précision de* 98.06% *dans un temps de 384s. a) Résultat final de la segmentation b) Histogramme avant reclassification finale montrant que la classification est déjà faite en* 8 *classes principales et que la reclassification est nécessaire pour reclasser les valeurs "hors classe" c) Segmentation de la sous-bande* V_3*. d) Histogramme de la sous-bande* V_3 *classifiée.*

10.1.4 Méthode par regroupement (K-means)

L'algorithme de la méthode "par regroupement" que nous utilisons est dérivé de la méthode K-means. Cette méthode effectue un partitionnement de l'image en K régions. A l'intérieur de chaque région de C_k pixels, on calcule la moyenne m_k et la variance v_k . Ensuite, pour chaque, on calcule soit la distance L_1 soit la distance L_2 définies par :

$$L_1(k) = \frac{|(D_{ij} - m_k)|}{\sqrt{v_k}} \tag{10.1}$$

et :

$$L_2(k) = \sqrt{\frac{\sum (D_{ij} - m_k)^2}{v_k}} \tag{10.2}$$

Ensuite le pixel D_{ij} est assigné à la classe k qui correspond à une distance minimale.

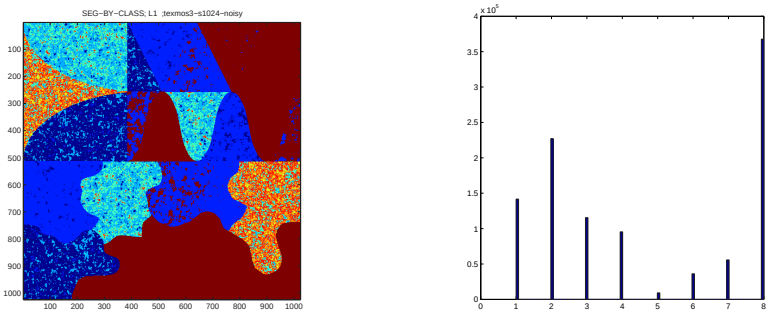


FIG. *10.6 – Résultats de classification par méthode de regroupement et les distances L_1 et L_2. Le seul paramètre est $K = 8$ classes requises (et obtenues d'après l'histogramme montré en b)). Pour les deux distances, le pourcentage de bonne classification est $P_a \simeq 50\%$ et le temps de calcul est 116s a) Résultat de la segmentation avec les distances L_1 et L_2. b) Histogramme de la segmentation finale en 6 classes.*

10.1.5 Résultats comparatifs

Afin de tester la qualité de la segmentation nous comptons le nombre $N_m(\boldsymbol{r})$ de pixels mal classifiés, dans l'image finale segmentée, par rapport à la segmentation z, connue, de l'image originale.
Nous pouvons comparer, dans le tableau qui suit ,les résultats de classification avec les trois méthodes et deux niveaux de décomposition différents pour la méthode WBPMS. Nous obtenons une classification très précise en comparaison de la méthode par regroupement et un temps de calcul beaucoup plus faible qu'avec la méthode de classification bayésienne dans l'espace direct (BPMS).

Méthode	Classes demandées/obtenues	Pixels mal-classifiés	Pourcentage de précision (P_a)	Tps de classif. total
BMPS (20 iter.)	8	206114	80.34%	1805s
WBMPS(J=4, 20 iter.)	8	55488	94.8%	260s
WBMPS(J=3, 20 iter)	8	20336	98.06%	384s
K-Means (L_1 and L_2)	8	530196	49.44%	116s

TAB. *10.1 – Comparaison des trois méthodes de segmentation sur la mosaïque de test "texmos3" de la base SIPI . L'image originale, quantifiée sur 8 bits, a été suréchantillonnée de 512×512 à 1024×1024 et a été perturbée par un bruit de variance différente pour chaque classe plus un bruit additif global gaussien. Le nombre de classes demandées est $K = 8$ pour toutes les méthodes, ce qui correspond au nombre de classes de l'image test. La qualité de segmentation est basée sur le nombre de pixels correctement classifiés, ou pourcentage de bonne classification P_a (percentage of accuracy), utilisé dans [SF03b, SF03a].*

10.2 Deuxième exemple

Dans ce deuxième exemple trois méthodes sont appliquées sur une image satellite naturelle (512×512 et 8 bits) représentant la côte de San Diego. Nous comparons les résultats de segmentation obtenus par les méthodes suivantes :
- la segmentation BPMS dans le domaine direct.
- la segmentation WBPMS dans le domaine des ondelettes.
- la segmentation semi-supervisée bayésienne basée sur un modèle HMT [CB01b].

Pour les deux premières méthodes nous avons imposé un nombre de classes $K = 6$. Le résultat avec le modèle HMT est tiré de [CB01b]. Il s'agit d'une segmentation en deux classes, donc fournissant une image binaire, et dont le but est de faire la discrimination entre les régions maritimes et les régions terrestres. Donc, premièrement, comme nous comparons les résultats sur des images naturelles, deuxièmement parce que nous effectuons notre test avec une valeur de classes supérieure à deux et troisièmement parce que la méthode HMT utilise un algorithme semi-supervisé basé sur la discrimination de textures, nous commenterons les résultats d'un point de vue qualitatif.

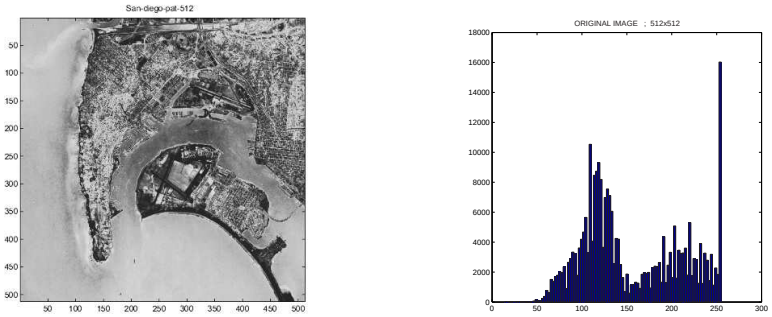


FIG. *10.7 – a) Image test de la côte San-Diego* 512×512, 8 *bits par pixel.) Histogramme montrant la loi de mélange en deux distributions, qui caractérisent les deux régions dominantes : terrestre et maritime.*

10.2.1 Méthode BPMS

La méthode BPMS a été testée avec $K = 6$ classes demandées et 20 itérations (Fig. 10.8). Le nombre de classes obtenu est 6. Le temps de calcul avec ces paramètres est 114s.

10.2.2 Méthode WBPMS

Sur l'image San Diego, la méthode WBPMS a été testée avec $K = 6$ classes demandées et 20 itérations comme pour la méthode BPMS (Fig. 10.9) . Le nombre de classes obtenu est 6. Le nombre d'échelles de la décomposition est 2. Le temps de calcul avec ces paramètres est de 79s ce qui est un peu plus rapide que la méthode BPMS (114s), pour le même nombre d'itérations. Ce résultat est dû principalement 1) au fait que l'image comporte beaucoup plus de détails que

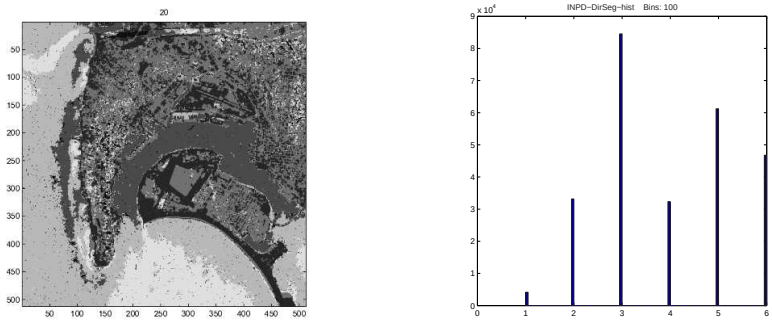


FIG. *10.8 – a) Segmentation bayésienne dans le domaine direct avec* $K = 6$ *classes et un nombre d'itérations* $itermax = 20$*. Temps de calcul* $114s$*. b) Histogramme de la segmentation finale.*

l'image de test texmos3s et donc que la segmentation dans les sous-bandes de détail est un peu plus longue 2) l'échelle maximale de décomposition est volontairement limitée à 2 pour éviter le risque de disparition des régions de petite taille et la sous-segmentation (nombre de classes trop faible dans la bande d'approximation ou nombre trop important de coefficients provenant des segments de détails mal classifiés). Nous n'avons pas effectué le test avec une image naturelle de plus grande taille qui permettrait d'augmenter le nombre d'échelles de décomposition.

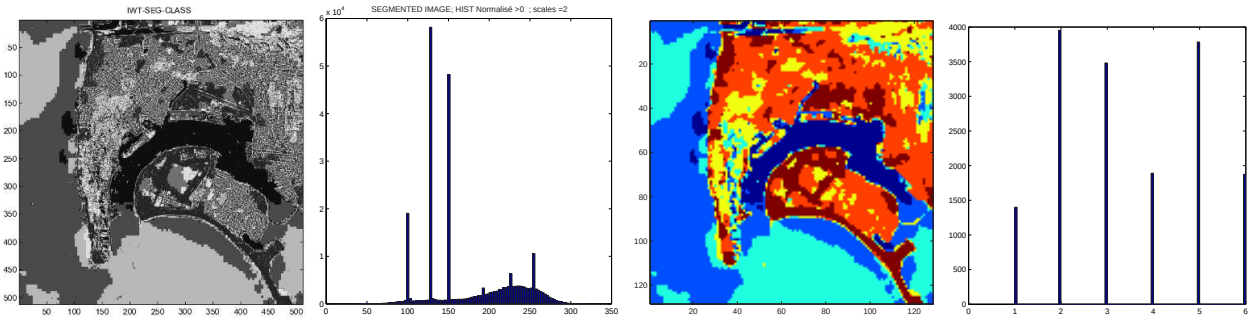


FIG. *10.9 – Segmentation WBPMS de l'image San-Diego. Le nombre de classes demandées et obtenu est* $K = 6$*. Le nombre d'itérations est* 20 *et le nombre d'échelles est* $J = 2$*. Le temps de segmentation est 79s. a) Segmentation finale b) Histogramme brut dans le domaine direct et avant reclassification, montrant* 6 *classes principales et de nombreuses valeurs d'étiquettes étalées entre ces classes principales. Ces valeurs d'étiquette proviennent essentiellement de la reconstruction des sous-bandes d'ondelette dont les coefficients sont modélisés par une gaussienne à forte variance. c) Segmentation de la sous-bande d'approximation* V_J*, à l'échelle la plus haute (basse résolution) d) Histogramme de la sous-bande d'approximation montrant la détection des* 6 *classes à ce niveau de la décomposition.*

10.2.3 Méthode HMT

Le but de la segmentation pour les auteurs était de séparer la zone terrestre de la zone maritime. Cette segmentation, basée sur un HMT, est donc réalisée avec deux classes. En utilisant une méthode HMT il est possible de tenir compte des textures. Celles-ci correspondent, dans le cas présent, aux textures des régions maritime et terrestre. La segmentation HMT non-supervisée a néanmoins dû être "initialisée" par un apprentissage de ces deux textures. Les auteurs ont pris deux sous-images de taille 100×100, dans deux coins de l'image complète originale (1024×1024), qui représentent le mieux les régions maritime et terrestre, puis ont déterminé quel modèle (matrices de transition inter-bandes, moyennes et variances) correspond le mieux à ces textures. Cette méthode présente donc la nécessité d'un apprentissage de modèles de textures puis du choix des paramètres de la texture observée en fonction de la corrélation avec un des modèles prédéfinis. En cela la méthode ne peut être considérée comme totalement non-supervisée. Ce serait le cas si le modèle utilisé savait détecter les différentes textures et leurs paramètres sans apprentissage sur une ou plusieurs zones de l'image. Nous pouvons d'autre part remarquer que les parties de la zone maritime (port submersible, droite supérieure des secondaire-images dans [CB01b]) sont réduites entre l'étape b) de la figure 10.10 et la fusion inter-échelles de l'étape et de la sous-image en c). Donc, si le résultat est bon pour la chaîne de montagne qui est classifiée dans comme zone terrestre, en revanche, certaines régions maritimes (ports de la baie) ont tendance à subir une réduction de taille. Nous avons d'ailleurs rencontré le même problème, avec notre méthode, en segmentant cette image en deux classes. Nous avons constaté que l'action sur les paramètres α_{Potts_a} et α_{Potts_d} dans les deux classes et de façon différente pour les sous-bandes d'approximation et de détail (augmentation ou diminution pour les deux types de sous-bandes ou variations opposées pour ces deux même types) favorise soit l'augmentation de la taille des régions maritimes soit l'augmentation de la taille des régions terrestres.

FIG. *10.10 – a) San Diego : image originale de* 1024×1024 *utilisée pour la phase d'apprentissage des deux textures (océan et terre), matérialisée par deux sous-images carrées de taille* 100×100. *b) Résultat d'une segmentation binaire brute au niveau pixel sur une sous-image de taille* 256×256. *c) Résultat de la segmentation binaire par fusion inter-échelles.*

10.2.4 Résultats

La segmentation est réalisée avec $K = 6$ classes.
- Méthode BPMS : Itermax = 20 ; Temps de calcul : 114.
- Méthode WBPMS : Itermax (Approx.) = 20 ; Itermax(Details) = 20 ; Le temps total de calcul est 79s avec une décomposition sur 2 échelles.
- Méthode HMT : nous ne connaissons pas le temps de calcul. Le résultat, avec cette méthode, prouve que les auteurs peuvent, après une phase d'apprentissage des textures maritimes et terrestres, de réaliser une bonne classification deux classes : l'une représentant les régions terrestres et l'autre représentant les régions maritimes. En particulier cette méthode a montré de bons résultats en segmentant la péninsule nord-sud ("point Loma"), région montagneuse et inhomogène, qui présente sur de nombreuses analogies avec les zones maritimes, en particulier parce qu'elle présente localement le même type de texture. En revanche, comme le montrent les résultats obtenus par les auteurs précités, l'étape de fusion inter-échelles qui a tendance à favoriser la croissance des régions terrestres pour une meilleure segmentation de la péninsule montagneuse, présente l'inconvénient de se faire au détriment des régions maritimes.

10.3 Troisième exemple

L'image test correspond à une bande spectrale d'une image satellite hyperspectrale "3D" composée de 244 bandes contiguës et déjà présentée dans la fig. 9.2. Le but ici est de nouveau la classification de façon totalement non-supervisée de cette image naturelle en régions homogènes et en détectant à la fois les éléments "de détail", notamment les routes, régions très étroites et d'orientations diverses, ainsi que les régions homogènes de beaucoup plus grande étendue.
Dans cet exemple nous avons encore pris le même nombre d'itérations pour les méthodes BPMS et WBPMS ,c'est-à-dire 20. Dans la méthode WBPMS, nous il nous est possible d'utiliser un nombre maximum d'itérations différent dans la sous-bande d'approximation et dans les sous-bandes de détail. Nous avons néanmoins gardé la même valeur $itermax = 20$ pour les deux types de sous-bandes de façon à ne pas raccourcir un peu "artificiellement" le temps de calcul par rapport à la méthode dans le domaine direct (BPMS). Cependant, le nombre d'itérations, selon les cas de figure, pourrait être réglé différemment dans les deux-types de sous-bandes. En effet la segmentation d'une image comportant de grandes régions très homogènes demande moins d'iterations dans les sous-bandes de détail qu'une image comportant de nombreuses discontinuités. Comme cette image est de taille 512×512, la méthode WBPMS est effectuée sur seulement deux niveaux de décomposition. Le temps de calcul avec la méthode WBPMS est de 73s contre $110s$ avec la méthode BPMS. Encore une fois ces images test ne montrent pas le mieux de la méthode dans le domaine des ondelettes puisque nous ne travaillons qu'avec deux niveaux de décomposition. Malgré cela, la figure ci-dessous (10.11) montre la meilleure qualité de segmentation, en seulement 20 itérations, de la méthode WBPMS par rapport à la méthode directe.

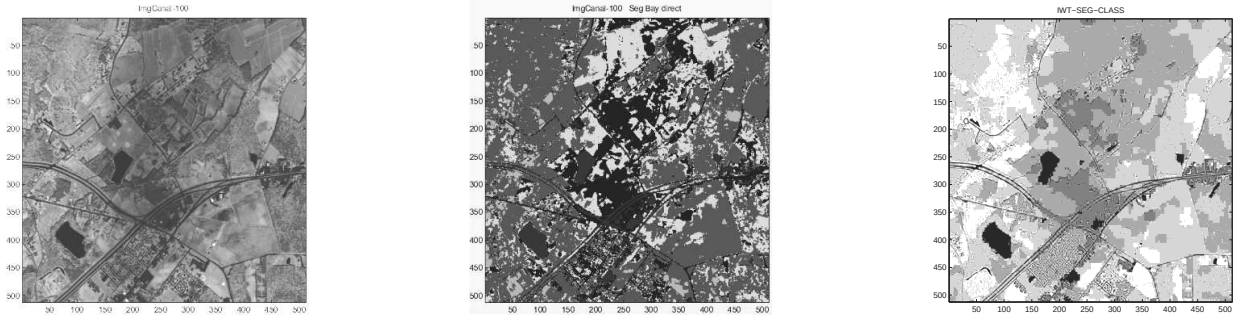


FIG. *10.11 – Comparaison des résultats de segmentation entre les méthodes BPMS et WBPMS sur une bande spectrale d'une image satellite hyperspectrale AVIRIS de taille* 512×512. *Les paramètres de la segmentation sont* $K = 6$ *classes demandées et* 20 *itérations (*20 *pour les sous-bandes V et W dans la méthode WBPMS). Les deux méthodes conduisent à une classification finale en* 6 *classes (nous ne montrons pas les histogrammes ici). a) Mono-bande originale de l'image hyperspectrale. b) Segmentation bayesienne dans le domaine direct (BPMS) ; le temps de segmentation est de* 110*s. c) Résultat de la segmentation avec la méthode WBPMS ; le nombre d'échelles est* 2 *et le temps de segmentation de* 73*s. Nous pouvons noter un temps de segmentation légèrement plus court pour la méthode WBPMS. Le résultat le plus important est la qualité de la segmentation obtenue en seulement* 20 *itérations avec la méthode WBPMS.*

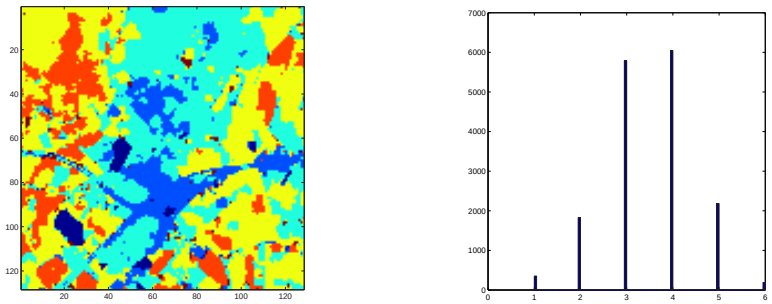


FIG. *10.12 – Détail de la segmentation de la sous-bande d'approximation* V_J *au niveau* $J = 2$ *et son histogramme montrant que le nombre de classes trouvées dans* V_J *correspond bien déjà au nombre final de classes requis et que la segmentation supplémentaire des sous-bandes de détails améliore considérablement la qualité de la segmentation finale.*

Conclusion à la seconde partie

Le Groupe des problèmes inverses (GPI), au LSS (Laboratoire des Signaux et Systèmes de Supélec), a développé un modèle de segmentation bayésienne basé sur une modélisation de Potts-Markov pour les étiquettes des pixels. Cet algorithme est mis oeuvre dans le domaine direct des pixels mais présente, nous l'avons vu dans le deuxième chapitre sur la segmentation de séquences, l'inconvénient majeur de nécessiter des temps de calcul quelquefois prohibitifs malgré la qualité du résultat obtenu. L'idée, dans de nombreux cas où la complexité est élevée, est de passer dans un domaine transformé qui, par exemple, réduit le nombre de données significatives et peut présenter d'autres propriétés intéressantes que nous pouvons utiliser comme a priori dans une approche bayésienne. L'analyse des propriétés statistiques des coefficients d'ondelettes d'une décomposition orthogonale rapide nous a conduit à choisir ce domaine pour adapter notre méthode de segmentation bayésienne directe au domaine transformé des ondelettes.

La première originalité de notre segmentation bayésienne dans le domaine des ondelettes réside dans le fait qu'elle ne repose pas sur une méthode par arbres de Markov cachés (HMT), qui utilise aussi largement les propriétés statistiques des coefficients d'ondelettes. Nous avons plutôt chercher à "projeter" notre méthode de segmentation bayésienne du domaine direct vers le domaine des ondelettes. Pour cela nous faisons des hypothèses de mélanges de gaussiennes pour les sous-bandes d'ondelettes et d'échelle et utilisons à nouveau un modèle de Potts-Markov pour la segmentation dans ces sous-bandes. Dans cette adaptation, une propriété majeure par rapport au domaine direct est que les segmentations, dans toutes les sous-bandes de détail, sont faites avec seulement deux classes. Nous obtenons ainsi, grâce à un modèle classique de mélanges de deux gaussiennes indépendantes pour les coefficients de détail, une réduction importante de la complexité. Les paramètres à estimer dans les itérations successives de l'échantillonnage de Gibbs ne concernent donc plus que deux gaussiennes pour toutes les sous-bandes de détail.

La deuxième originalité de ce travail est que notre modèle de Potts-Markov dans le domaine direct a été étendu, pour les ondelettes, à deux voisinages du premier et du second ordre (8-connexité) pour les étiquettes des pixels. Ce modèle permet en effet de tenir compte des orientations privilégiées des trois sous-bandes de détail de la TO. Ce schéma de segmentation PMRF-ondelettes a conduit à une réduction du temps de segmentation pouvant atteindre un facteur dix, voire davantage dans le cas d'images de grande taille (supérieures à 512×512), tout en conservant une qualité élevée. Ce schéma

a été testé avec plusieurs niveaux de décomposition de la TO. Nous avons obtenu des résultats optimaux avec un nombre maximum de niveaux de décomposition de deux ou trois pour la plupart des images synthétiques et naturelles. En revanche, lorsque la taille de l'image augmente, le nombre d'échelles de décomposition peut être augmenté et la vitesse de segmentation est alors nettement réduite par rapport aux autres méthodes. De plus, la qualité de la segmentation pour des images bruitées a montré de très bons résultats. Il faut rappeler que nos méthodes de segmentation, aussi bien dans le domaine direct que dans celui des ondelettes, prennent en compte un bruit gaussien de moyenne et de variance quelconques et que ces paramètres peuvent changer dans les différentes régions de l'image. Les paramètres de bruit sont estimés automatiquement lors des itérations de Gibbs, à la fois globalement, c'est-à-dire pour toute l'image, et localement pour chacune des régions. Dans ce sens nous pouvons dire que nous avons atteint notre but qui est de présenter une méthode de segmentation de bonne qualité, rapide, non-supervisée et pour des images fortement bruitées.

Cette segmentation "combinée" bayésienne-ondelettes trouve une première application dans la segmentation et dans la compression rapide d'images fixes. Une deuxième application est la segmentation de séquences vidéo, initiée dans le chapitre deux, ainsi que l'estimation de mouvement dans les nouveaux schémas de compression que nous avons développés [5] et [6], mais aussi dans des applications médicales d'analyse du mouvement ou post-acquisition d'estimation, d'analyse de mouvement et de compression. En effet nous ne prétendons pas, par ces méthodes, nous rapprocher des méthodes de réduction de redondance temporelle et d'E.M. rapides adaptées à la vidéo dites "temps réel". Notre but est de fournir un outil rapide de segmentation, voire d'estimation et d'analyse de mouvement, pour des applications nécessitant une qualité importante en terme de segmentation et d'analyse du mouvement.

Enfin l'ensemble des travaux de cette deuxième partie a donné lieu aux publications et présentations [5, 4, 3, 2].

Conclusion générale

Cette thèse vous a présenté deux développements en principe bien distincts du traitement du signal : l'estimation de mouvement et la segmentation. Cependant au cours du développement de ces deux parties, nous avons mis l'accent sur les points communs à ces deux parties et sur le "chaînon" qui les rattache.

Dans la première partie nous avons montré que l'estimation de mouvement prise dans un sens contextuel et non brut, tel que le calcul de vecteurs de mouvement sur des blocs d'image, apporte des informations plus précises qui permettent de réaliser des opérations telles que l'analyse du mouvement sur des objets ou des régions d'une scène et la segmentation de ces objets. Elle conduit aussi à l'identification de trajectoires des objets ainsi qu'à l'analyse de scène. C'est cette approche contextuelle des scènes que nous avons mise en avant car elle semble indissociable à terme des méthodes de compression complexes et "intelligentes". Pour réaliser la mise en oeuvre de l'analyse du mouvement, nous nous sommes orientés vers des familles d'ondelettes redondantes peu usitées et offrant cependant une alternative aux méthodes de résolution du flot optique pour une analyse relativement précise du mouvement dans les séquences d'images. Nous avons finalement montré comment un schéma, basé sur la mesure des paramètres de mouvement d'objets acquis au moyen de ces familles d'ondelettes adaptées au mouvement, pouvait être exploité pour modéliser les trajectoires des objets et réaliser une estimation de mouvement basée sur les trajectoires de ces mêmes objets. Enfin il nous semble intéressant de considérer que lors d'approches telles que la compréhension avancée de scènes (de circulation routière par exemple), l'analyse de la scène conduit, a priori, à déterminer les mouvements de divers objets ou personnages. La modélisation, et la compréhension, de ces mouvements doit donc pouvoir être reprise en compte dans des schémas d'estimation de mouvement pour la compression des flux vidéo de ces mêmes scènes. Ainsi l'analyse de scène conduirait naturellement à une compression évoluée et contextuelle des séquences d'images.

Dans la seconde partie nous avons abordé le problème de la classification/segmentation entièrement non supervisée pour des images fixes et pour des séquences d'images. Le premier avantage d'une bonne classification en régions homogènes est de permettre la simplification de l'image en un nombre limité de régions étiquetées. Le nombre d'étiquettes fixe le nombre de symboles pour la quantification. Ainsi une image segmentée en 8 étiquettes peut se représenter par seulement 3 bits de quantification, ce qui réduit considérablement la taille du débit binaire. Cet aspect est particulièrement intéressant dans un contexte de compression orienté aussi "objet" ou régions qui sont les parties qui nous intéressent par opposition aux "textures" de ces régions. Nous avons montré que la seg-

mentation en régions permet de réaliser l'estimation du mouvement des régions par mesure du déplacement des centres de masse de ces régions. Dans une deuxième étape nous avons développé un algorithme de segmentation/classification dans le domaine des ondelettes, ce qui a permis de réduire les temps de segmentation de façon très significative par rapport à l'algorithme basé sur un modèle équivalent dans le domaine direct.

Enfin, les publications suivantes : [12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2] ont été réalisées durant cette thèse.

Annexe partie I

Définition 3 (Décomposition continue en ondelettes réelles) *La transformée en ondelettes d'une fonction $f(x)$ sur une famille d'opérateurs $\psi_{b,a}$, où b et a représentent respectivement la translation et le changement d'échelle, s'exprime par :*

$$Wf(b,a) =< f, \psi_{b,a} >= \int_{-\infty}^{+\infty} f(x) \frac{1}{\sqrt{a}} \psi^*(\frac{x-b}{q}) dx \tag{10.3}$$

avec $f \in \mathbf{L}^2(\mathbb{R})$

Théorème 4 (Admissibilité. Calderon, Grossman, Morlet) *Une transformée en ondelettes est complète et préserve l'énergie si elle satisfait la condition d'admissibilité sur l'ondelette définie par :*

$$c_\psi = 2\pi^3 \int_{\mathbb{R}^2} \int \mathbb{R} \frac{|\hat{\psi}(\vec{k}, \omega)|^2}{|\vec{k}|^2 |\omega|} d^2\vec{k} d\omega < \infty \tag{10.4}$$

où $\psi \in \mathbf{L}^2(\mathbb{R})$. Cette condition traduit, pour l'ondelette, 1) que sa moyenne est nulle et 2) qu'elle est indéfiniment dérivable sur $\mathbf{L}^2(\mathbb{R})$

Définition 5 (Complétude) *La décomposition d'une fonction f par une famille d'opérateurs satisfaisant la condition d'admissibilité est complète si cette fonction peut se recomposer intégralement à partir de sa décomposition.*

$$f(x) = \frac{1}{c_\psi} \int_0^{+\infty} \int_{-\infty}^{+\infty} Wf(a,b) \frac{1}{\sqrt{a}} \psi(\frac{x-b}{a}) db \frac{da}{a^2} \tag{10.5}$$

Propriété 6 (Conservation de l'énergie)

$$\int_{-\infty}^{+\infty} |f(x)|^2 dx = \frac{1}{C_\psi} \int_0^{+\infty} \int_{-\infty}^{+\infty} |Wf(b,a|^2 db \frac{da}{a^2} \tag{10.6}$$

Glossaire partie I

- 4CIF Four times the surface of the CIF image format : 704 x 576 pixels
- AAL ATM Adaptation Layer
- ACP Analyse en Composantes principales
- AVC Advanced Video Coding. Nom donné à la nouvelle norme H264 ou MPEG4, part 10 de la JVT.
- BAP Body Animation Parameter
- BDP Body Definition Parameter
- BIFS Binary Format for Scenes
- BM Block Matching. Technique d'estimation de mouvement par de comparaison de blocs
- CABAC Context-based Adaptive Binary Arithmetic Coding
- CAVLC Context-based Adaptive Variable Length Coding
- CIF Common Interchange Format : 352x288 pixels (voir QCIF, version CIF redimensionnée à un quart de cette taille)
- CNC Continuous Normalized Convolution : extension of normalized convolutions for irregularly sampled signals ([And02]
- CT Curvelet Transform
- CWT Continuous Wavelet Transform (La transformée continue en ondelettes est très souvent assimilée à la transformée dyadique de Mallat car elle est "continue" (non-décimée) en position. Cependant le terme "transformée continue" ne devrait être théoriquement utilisé que pour décrire la version sous forme d'une intégrale continue, de la transformation en ondelettes. Patrice Abry [Abr97] utilise un terme beaucoup moins ambigu à la place du terme "transformée dyadique" de Mallat : la CDWT, ou Continuous-Discrete (ou -Dyadique) Wavelet Transform signifie bien que cette transformée est (pseudo-)continue en position, c'est-à-dire non-décimée, et dyadique en échelle.
- DDM Duval-Destin Murenzi (Separable, Motion-Tuned, CWT)
- DVC "DWT Video Coder" : codeur vidéo par ondelettes proposé pour MPEG4
- DWT Discrete wavelet transform. Terme souvent utilisé à la place de transformée orthogonale, la DWT est plus exactement la version échantillonnée, donc la version sous forme de somme discrète, de l'expression intégrale de la T.O.
- EM Estimation de mouvement.
- ES Elementary Stream : Une séquence de données dont l'origine est un seul producteur dans le terminal de transmission MPEG4 et dont la destination est un récepteur unique, par exemple un objet ou une entité de contrôle dans le terminal MPEG4. Transite dans un canal FlexMux.
- EQM voir MSE
- EBCOT Embedded Coding With Optimized Truncation
- ESCOT Three Dimensional Embedded Subband Coding With Optimized Truncation (3-D ESCOT)
- EZW Embedded Zero-Tree Wavelet
- GMC Global Motion Compensation
- GOF Group Of Frames

- GOP Group Of Pictures
- HCR Huffman Codeword Reordering
- HT Hadamard Transform
- JVT Joint Video Team : nom du regroupement des organismes de normalisation ISO (MPEG) et ITU (H26x) pour créer le standard H264.
- KLT Karhunen-Loewe Transform (voir ACP)
- LPC Linear Predictive Coding
- LSP Line Spectral Pairs
- LTP Long Term Prediction
- MCTF Motion Compensated Temporal Filtering
- MHME Multiple Hierarchical Estimation ([And02], section 5.7)
- MPEG Moving Pictures Experts Group
- MPEG1 Norme de compression video sur CDROM. Débit $1,5$ Mbits/s
- MPEG2 Norme de compression video en mode "broadcast" (télévision). Débit $1,5$ à 30 Mbits/s
- MPEG4 Norme de compression basée sur MPEG2 avec, entre autres, le codage indépendant des objets vidéo (VO).
- MSE Mean Square error, ou EQM (erreur quadratique moyenne).

$$MSE = \sum_{n,m} (I_g(n,m) - I_f(n,m))^2$$

  où I_g est l'image traitée (ou observée) et I_f est l'image d'origine.
- MTSTWT Motion-Tuned Spatio-Temporal Wavelet Transform
- NAL Network Abstraction Layer
- OD Object descriptor
- OBMC Overlapped Block Motion Compensation
- OWT Orthogonal Wavelet Transform
- PCA Principal Components Analysis (voir ACP)
- PSNR Power of Signal to Noise Ratio : mesure de qualité par la puissance du rapport signal/ bruit
  Pour une image (2D) :

$$PSNR_{dB} = 10\ log_{10} \frac{\sum_{n,m} I_f^2(n,m)}{\sum_{n,m} (I_g(n,m) - I_f(n,m))^2}$$

  où I_g est l'image observée (bruitée), I_f est l'image d'origine, ϵ est le bruit ($\epsilon = I_g - I_f$) et (n,m) sont les coordonnées des sites de l'image.
  Pour les signaux aléatoires, on définit le PSNR par :

$$PSNR_{dB} = 10\ log_{10} \frac{\sigma_f^2}{\sigma_\epsilon^2}$$

- QCIF Quarter Common Intermediate Format : 176x144 pixels
- RD Radon Transform
- ROI Region of Interest
- RT Ridgelet Transform
- RVLC Reversible Variable Length Coding
- SA-DCT shape-adaptive DCT
- SIF Standard Input Format.
- SNHC Synthetic-Natural Hybrid Coding
- SPIHT Set Partioning In Hierarchical Trees.
- SQCIF Sub-QCIF : 128x96 pixels

- ST Domaine “Spatio Temporel”. Se dit d’une application dans le domaine spatial + temporel. Pour ce qui est des dimensions, notre application se fait dans un domaine ST en trois dimensions : 2D pour le spatial et 1D pour le temporel. Mais dans le cas d’une analyse spatiale volumique, il s’agira de 3D+T. De plus nous avons précisé lorsqu’il s’agit du domaine ST “direct”, c’est-à-dire directement dans le domaine spatial + temporel ou lorsqu’il s’agit de la version “spectrale” du domaine ST, donc selon les vecteurs d’onde spatiaux notés k_x et k_y et temporel k_t (ou pulsation ω).
- STFT Short term Fourier transform. Transformée de Fourier à fenêtre.
- SVD Singular Values Decomposition (voir ACP)
- Sprite Un ”sprite” représente en général tout ou partie du fond d’image pas, ou peu, animé.
- TO Transformée en ondelettes
- VLBV Very Low Bitrate Video
- VO Video Object
- VOP Video Object Plane (I-, P-, B-VOP). Occurence d’un VO à un instant t précis ([Ric03], §5.2).
- XMT Extensible MPEG-4 textual format
- Y4M YUV for MPEG format scaler. Extension des fichiers YUV dont l’échelle a été changée.

Glossaire partie II

- AIC : Akaike Information Criterion
- BBSS : Bayesian Blind Source Separation
- BIC : Bayesian Information Criterion
- BF : Bayes Factor
- BPMS : Bayesian Potts-Markov Segmentation
- CT : Computed Tomography
- EB : Empirical Bayes
- ELQM : Estimation Lineaire en Moyenne Quadratique
- EM Expectation Maximization (Maximum d'Espérance)
- GPAV52S (seismic) : Code de calcul de Gradient développé au GPI (avec initialisation du gradient, choix de paramètres ; sous Matlab).
- GMM Gaussian Mixture Model
- GNC Graduated non Convexity. [Blake et Zisserman 87]
- HMM Hidden-Markov Model (Modèle de Markov caché)
- HMT Hidden-Markov Tree (Arbre de Markov caché)
- IID Independent and identically distributed
- ICE Iterative Conditional Expectation (Espérance Conditionnelle Itérative)
- ICM Iterative Conditional Modes
- IGMM Independent Gaussians Mixture Model
- IRM (MRI) Imagerie par Résonance Magnétique
- MAP Maximum a posteriori
- MCMC Markov Chain Monte-Carlo
- MEG/EEG Magneto/Electro Encephalography
- MMTO Méthode des maxima de la transformée en ondelettes (WTMM). Voir [AAE$^+$95, MZ92, MH92]
- MPM Maximum a posteriori Marginal
- MRI (voir IRM)
- PM Posterior Mean ; Moyenne a posteriori
- PMRF Potts-Markov Random Field.
- SEL Sequential Edge Linking (Sequential Search Algo [Eichel et al.])
- SMG Scale Mixture of Gaussians.
- SSM Strict Sense Markov
- WBPMS : Wavelet-Bayesian Potts-Markov Segmentation

– WSM Wide Sense Markov
– WTMM Wavelet transform maxima method (voir MMTO)

Bibliographie partie I

[AAE+95] A. Arneodo, F. Argoul, Bacry E., Elezgaray J., and Muzy J.F. *Ondelettes, multifractales et turbulences ; de l'ADN aux croissances cristallines*. Diderot, Sciences en actes, 1995.

[AB85] E. H. Adelson and J. R. Bergen. Spatiotemporal energy models for the perception of vision. *Journal of Optical Society of America*, A2 :284–299, February 1985.

[Abr97] P. Abry. *Analyse continue par ondelettes*. Diderot Sciences en Actes, 1997.

[AM96] J.P. Antoine and R. Murenzi. Two-dimensional directional wavelets an the scale-angle representation. *Signal Processing*, 52 :259–281, 1996.

[AM98] J.P. Antoine and R. Murenzi. Galilean wavelets : Coherent states of the affine galilei group. Technical report, UCL, Université Catholique de Louvain-la-Neuve, 1998.

[Amb00] S. Ambellouis. *Analyse du mouvement dans les séquences d'images par une méthode récursive de filtrage spatio-temporel sélectif*. PhD thesis, Université des Sciences et Technololgies de Lille, 2000.

[Amo04] I. Amonou. *Thèse : Décompositions Hiérarchiques Non-Redondantes Orientées Régions pour le Codage des Images Numériques*. Ecole doctorale d'informatique, télécommunications et electronique de paris ; spécialité signal et images, Ecole Nationale Supérieure des Télécommunications, 2004.

[AMV99] J.P. Antoine, R. Murenzi, and P. Vandergheynst. Directional wavelets revisited : Cauchy wavelets and symmetry detection in patterns. *Applied an Computational Harmonic Analysis*, 6 :314–345, 1999.

[AMVA04] J.P. Antoine, R. Murenzi, P. Vandergeynst, and J.P. Ali. *2D CWT Wavelets*. Cambridge Press, UCL, Université Catholique de Louvain-la-Neuve, first edition, 2004.

[And02] K. Andersson. *Quality and Motion Estimation for Image Sequence Coding*. PhD thesis, Linköping Universitet, Sweden, February 2002.

[Bac03] E. Bacry. *LastWave Manual and C-Code*. CMAP, Centre de mathématiques appliquées, Ecole Polytechnique Palaiseau, 98 and 2003 edition, 2003.

[Bal98] M. Balsi. Focal-plane optical flow computation by foveated CNNs. In *Proc. of Fifth IEEE International Workshop on Cellular Neural Networks and their Applications (CNNA-98)*, volume A2, pages 149–154, London, UK, April 1998.

[BB00] S. Beauchemin and J. Barron. The Fourier properties of discontinuous motion, 2000.

[BBAT97] G.D. Borshukov, G. Bozdagi, Y. Altunbasak, and A.M. Tekalp. Motion Segmentation by Multistage Affine Clasification. *IEEE Transactions on Image Processing*, 6(11) :1591–1594, 1997.

[Ber99a] C.P. Bernard. Discrete wavelet analysis for fast optic flow computation. Rapport Interne du Centre de Mathématiques Appliquées RI415, Ecole Polytechnique, CMAP, Centre de Mathématiques Appliquées, Palaiseau, France., Février 1999.

[Ber99b] C.P. Bernard. *Thèse : Ondelettes et problèmes mal posés : la mesure du flot optique et l'interpolation irrégulière.* PhD thesis, Ecole Polytechnique, CMAP, Centre de Mathématiques Appliquées., Palaiseau, France, Novembre 1999.

[Bey92] G. Beylkin. On the representation of operators in bases of compactly supported wavelets. *SIAM J. on Numerical Analysis*, 6(29) :1716–1740, 1992.

[BF98] P. Bouthemy and R. Fablet. Motion characterization from temporal cooccurrences of local motion-based measures for video indexing. In *ICPR, Proc. of Int'l. Conf. on Pattern Recognition.*, Brisbane, Australia, August 1998.

[BFB94] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of optical flow techniques. *IJCV International Journal of Computer Vision*, 12(1) :43–77, 1994.

[BH95] B. Burke-Hubbard. *Ondes et ondelettes. La saga d'un outil mathématique.* Belin, collection "pour la science", 1995.

[BHY00] D. Bereziat, I. Herlin, and L. Younes. A generalized optical flow constraint and its physical interpretation. In *Proceedings of CVPR*, 2000.

[BJ97] K. Berkner and R.O.Wells Jr. A fast approximation of the continuous wavelet transform with applications. In *Proceedings of Asilomar*, 1997.

[BLJ95] L. Bonnaud, C. Labit, and Konrad J. Interpolative coding of image sequences using temporal linking of motion-based segmentation. In *ICASSP*, 1995.

[BM01a] P. Brault and H. Mounier. Automated, transformation invariant, shape recognition through wavelet multiresolution. In *Proceedings of the SPIE, International Society for Optical Engineering Wavelets : Applications in Signal and Image Processing IX, Andrew F. Laine ; Michael A. Unser ; Akram Aldroubi ; Eds.*, volume 4478, pages 434–443, San Diego, 2001.

[BM01b] P. Brault and H. Mounier. Wavelet multi-resolution transform applied to shape recognition based on a curvature criterion. In *Proceedings of the IAPR International Conference on Image and Signal Processing*, Agadir, 2001.

[Bra03a] P. Brault. Motion estimation and video compression with spatio-temporal motion-tuned wavelets. *WSEAS Transactions on Mathematics*, 2(1 & 2) :67–78, 2003.

[Bra03b] P. Brault. A new scheme for object-oriented video compression and scene analysis, based on motion tuned spatio-temporal wavelet family and trajectory identification. In *Proceedings of the 3rd IEEE International Symposium on Signal Processing and Information Technology*, Darmstadt, 2003.

[Bra04] P. Brault. On the performances and improvements of motion-tuned wavelets for motion estimation. *WSEAS Transactions on Electronics*, 1(1) :174–180, 2004.

[Bri87] W.L. Briggs. Multigrid tutorial. *SIAM*, 1987.

[BRS+94] T.J. Burns, Rogers, S.K., D.W. Ruck, and M.E. Oxley. Discrete, spatiotemporal, wavelet multiresolution analysis method for computing optical flow. *Optical Engineering*, 33(7) :2236–2247, July 1994.

[BSB03] P. Brault, J.L. Starck, and P. Beauvillain. Characterization of nanostructures stm images with the wavelet and ridgelet transforms. In *Proceedings of the IAPR International Conference on Image and Signal Processing*, Agadir, 2003.

[BSJ99] C. Bernard, Mallat S., and Slotine J.J.E. Wavelet interpolation networks for hierarchical approximation. In *Proc. of SPIE's 44th Annual Meeting*, Denver, CO., 1999.

[BV03] P. Brault and M. Vasiliu. Motion-compensated spatio-temporal filtering with wavelets. *WSEAS Transactions on Computers*, 2(4) :1131–1140, 2003.

[Can98] E. J. Candès. *Ridgelets : Theory and Applications*. Ph.d. thesis, Department of Statistics, Stanford University., 1998.

[Can99a] E. J. Candès. Harmonic analysis of neural networks. *Appl. Comput. Harmon. Anal.*, 6 :197–218, 1999.

[Can99b] E. J. Candès. Ridgelets : Estimating with ridge functions. Technical report, Department of Statistics, Stanford University, Submitted to Ann. Statist., 1999.

[CH99] J.K. Chang and T. L. Huntsberger. Dynamic motion analysis using wavelet flow surface images. *Pattern Recognition Letters*, 20(4) :383–393, 1999.

[CHA03] P. Carre, D. Helbert, and E. Andres. 3d fast ridgelet transform. *Proceedings of ICIP*, 1 :1021–4, Septembre 2003.

[CLF95] Clifford, K. Langley, and D.J. Fleet. Centre-frequency adaptive IIR temporal filters for phase-based image velocity estimation. *Image Processing and its Applications*, 4(6) :173–177, 1995.

[CLK99] J. Corbett, J.-P. Leduc, and M. Kong. Analysis of deformational transformations with spatio-temporal continuous wavelet transforms. In *Proceedings of IEEE-ICASSP*, page 4, Phoenix,AZ, 1999.

[CP03] N. Cammas and S. Pateux. Codage video scalable par maillage et ondelettes 3d. In *Proceedings of CORESA*, 2003.

[CTS97] M. M. Chang, A. M. Tekalp, and M. I. Sezan. Simultaneous motion estimation and segmentation. *IEEE Trans. Image Processing*, 6(9) :1326–1333, 1997.

[CVZ98] B. Chai, J. Vass, and X. Zhuang. Statistically adaptive wavelet image coding, 1998.

[Dau92] I. Daubechies. *Ten Lectures on Wavelets*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992.

[DBB91] J.N. Driessen, L. Boroczky, and J. Biemond. Pel-recursive motion field estimation from image sequences. *Journal of Visual Communication and Image Reproduction*, 2 :259–280, 1991.

[DC98] F. De Coulon. *Théorie et traitement des signaux*, volume VI of *Traité d'Electricité ; direction J. Neyrinck*. Presses Polytechniques Romandes, Ecole Polytechnique Fédérale de Lausanne, 1998.

[DDM93] M. Duval-Destin and R. Murenzi. *Progress in Wavelet Analysis and Applications*, chapter Spatio-temporal wavelets : Applications to the analysis of moving patterns, page 399408. Frontières, Gif-sur-Yvette,France, 1993.

[Del97] B. Delamotte. *Un soupçon de théorie des groupes : groupe des rotations et groupe de Poincaré*. DEA "Champs, Particules, Matière, LPTHE, Université Paris VII, Université Paris VI, 1997.

[Dir58] P.A.M. Dirac. *The Principles of Quantum Mechanics*. The international series and monographs on physics 27. Oxford Science Publications, 4eme edition, 1958.

[DKA95] R. Deriche, P. Kornprobst, and G. Aubert. Optical-flow estimation while preserving its discontinuities : A variational approach. In *Proc. Second Asian Conf. Computer Vision,ACCV*, volume 2, pages 290–295, Singapore, December 1995.

[DM95] F. Dufaux and F. Moscheni. Segmentation-based motion estimation for second generation video coding techniques. Technical report, MIT, Media Lab. and Swiss FIT, S.P. Lab, 1995.

[DML95] F. Dufaux, F. Moscheni, and A. Lippman. Spatio-temporal segmentation based on motion and static segmentation. *ICIP*, pages 306–309, 1995.

[DMZ96] X. Descombes, R. Morris, and Zerubia. Quelques améliorations à la segmentation d'images bayésiennes. Rapport de recherche 2916, INRIA, juin 1996.

[DMZB99] X. Descombes, R. Morris, J. Zerubia, and M. Berthod. Estimation of markov random field prior parameters using markov chain monte carlo maximum likelihood. *IEEE Trans. Image Processing*, 8(7) :954–963, July 1999.

[DS98] I. Daubechies and W. Sweldens. Factoring wavelet transforms into lifting steps. *J. Fourier Anal. Appl.*, 4(3) :245–267, 1998.

[Dut89] P. Dutilleux. *Wavelets Time-Frequency Methods and Phase Space*, chapter an Implementation of the algorithme a trous to compute the wavelet transform, pages 298–304. Springer-Verlag, Berlin, 1989.

[FA91] W.T. Freeman and E.H. Adelson. The design and use of steerable filters. *IEEE transactions on PAMI*, 13 :891–906, 1991.

[Fey79] R. Feynman. *Le cours de physique de Feynman : Mécanique Quantique*. InterEditions, edition originale : the feynman lectures on physics, 1965, caltech edition, 1979.

[FJ90] D.J. Fleet and A.D. Jepson. Computation of component image velocity from local phase information. *International Journal of Computer Vision*, 5(77–104), 1990.

[FSR03] M. Fliess and Hebertt Sira-Ramirez. An algebraic framework for linear identification. In *ESAIM Contr. Opt. Calc. Variat.*, volume 9, 2003.

[GKMM89] A. Grossman, R. Kronland-Martinet, and J. Morlet. *Wavelets Time-Frequency Methods and Phase Space. Proceedings of the International Conference of Marseille, dec. 87*, chapter Introduction to Wavelet Transforms, pages 2–20. Springer-Verlag, Berlin, 1987-89.

[Gui94] A. Guichardet. *Majeure de mathématiques. Groupes de Lie, représentations*. Département de mathématiques, Ecole Polytechnique Palaiseau, 1994.

[Gui96] J.P. Guillois. *Tehniques de Compression des Images*. Hermes, 1996.

[HCPS05] L. Hong, N. Cui, M. Pronobis, and S. Scott. Local motion feature aided ground moving target tracking with gmti and hrr measurements. *IEEE Transactions on Automatic Control*, 50(1) :127–133, january 2005.

[Hee87] D.J. Heeger. Optical flow using spatiotemporal filters. *International Journal of Computer Vision*, 1 :279–302, 1987.

[HKMMP88] M. Holschneider, R. Kronland-Martinet, J. Morlet, and P.Tchamitchian. The à trous algorithm. In *CPT-88/P.2215*, pages 1–22, Berlin, 1988.

[HS81] B.K.P Horn and B.G. Schunck. Determining optical flow. *Artificial Intelligence*, 17 :185–204, 1981.

[JBBA00] S. Jehan-Besson, M. Barlaud, and G. Aubert. Segmentation et suivi des objets en mouvement dans une séquence vidéo par contours actifs basés régions. In *Procesdings of CORESA*, Poitiers, octobre 2000.

[JMR00] S. Jaffard, Y. Meyer, and R.D. Ryan. *Wavelets : Tools for Science and Technology*. SIAM, 2000.

[KD92] J. Konrad and E. Dubois. Bayesian estimation of motion vector fields. *PAMI*, 14 :910–927, 1992.

[LCK+98] J.P. Leduc, J. Corbett, M. Kong, V.M. Wickerhauser, and B.K. Ghosh. Accelerated spatio-temporal wavelet transforms : An iterative trajectory estimation. In *Proceedings of ICASSP*, pages 2777–2780, 1998.

[Led94] J.P. Leduc. *Digital Moving Pictures*, volume 3 of *Advances in Image Communication*. Elsevier, 1994.

[Led97] J.P. Leduc. Spatio-temporal wavelet transforms for digital signal analysis. *Signal Processing*, 60 :23–41, 1997.

[Lee04] C.M. Lee. *Joint source-Channel Coding Tools for Robust Transmission of Video Sequences ; Application to H.263+ and H.264*. PhD thesis, LSS, Laboratoire des Signaux et Systèmes, Supélec, Université Paris-Sud, 2004.

[LHHC96] H. Liu, T.H. Hong, M. Herman, and R. Chellappa. Accuracy vs efficiency trade-offs in optical flow algorithms. In *Proceedings of European Conference on Computer Vision*, volume II, pages 174–183. Springer-Verlag, April 1996.

[LK81] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedgins of DARPA Image Understanding Worshop*, pages 121–130, 1981.

[LL63] J.M. Levy-Leblond. Galilei group and non-relativistic quantum mechanics. *Journal of mathematical physics*, 4(6), 1963.

[LMMS97] J.-P. Leduc, F. Mujica, R. Murenzi, and M.J.T. Smith. Spatio-temporal wavelet transforms for motion tracking. In *Proceedings of ICASSP*, volume 4, pages 3013–3016, 1997.

[LMMS00] J.P. Leduc, F. Mujica, R. Murenzi, and M.J.T. Smith. Spatiotemporal wavelets : A group-theoretic construction for motion estimation and tracking. *Siam Journal of Applied Mathematics*, 61(2) :596–632, 2000.

[LOL97] J.-P. Leduc, J.-M. Odobez, and C. Labit. Adaptive motion-compensated wavelet filtering for image sequence coding. *IEEE Transactions on Image Processing*, 6(6) :862–878, 1997.

[LP02] E. Le Pennec. *Bandelettes et représentation géometrique des images*. PhD thesis, Ecole Polytechnique, Palaiseau, 2002.

[LPM00] E. Le Pennec and S. Mallat. Image compression with geometrical wavelets. *Proceeedings of ICIP*, 2000.

[LPM03] E. Le Pennec and S. Mallat. Représentation d'image par bandelettes et application à la compression. *Proceeedings of GRETSI*, 2003.

[LZ93] B. Liu and A. Zaccarin. New fast algorithms for the estimation of block motion vectors. *IEEE Transactions on Circuits, Systems and Video Technology*, 3 :148–157, 1993.

[MAG88] S.A. Mahmoud, M.S. Afifi, and R.J. Green. Recognition and velocity computation of large moving objects in images. In *ICASSP*, volume 36, pages 1790–1791, 1988.

[Mal89] S. Mallat. A theory for multiresolution signal decomposition. *IEEE Transactions on PAMI*, 11 :167–180, 1989.

[Mal00] S. Mallat. *Majeure de mathématiques appliquées : traitement du signal*. CMAP, Département de mathématiques appliquées, Ecole Polytechnique Palaiseau, 2000.

[Mal01] S.G. Mallat. *A Wavelet Tour of Signal Processing*. Addison Westley, 1 and 2 edition, 1997 and 2001.

[MB94a] F.G. Meyer and P. Bouthemy. Region-based tracking using affine motion models in long image sequences. *CVGIP : Image Understanding*, 60(2) :119–140, September 1994.

[MB94b] D. Murray and A. Basu. Motion tracking with an active camera. *IEEE PAMI*, 16 :449–459, 1994.

[MCWP96] M.K. Mandal, E. Chan, X. Wang, and S. Panchanathan. Multiresolution motion estimation techniques for video compression. *Optical Engineering*, 35(1) :128–136, January 1996.

[MDK95] F. Moscheni, F. Dufaux, and M. Kunt. A new two-stage global/local motion estimation based on a background/foreground segmentation. In *IEEE proceedings of ICASSP*, Detroit, Michigan, May 1995.

[MH92] S.G. Mallat and W.L. Hwang. Singularity detection and processing with wavelets. *IEEE Transactions on Information Theory*, 38(2) :617–643, 1992.

[MHCP00] D. Marpe, G. Heising, H.L. Cycon, and A. Petukhov. Wavelet-based video coding using image warping and overlapped block motion compensation. In *IEEE transactions on Circuits and Systems for Video Technology*, 2000.

[MK96] J.F.A. Magarey and N.G. Kingsbury. An improved motion estimation algorithm using complex wavelets. In *Proceedings of ICIP*, pages 969–972, Lausanne, September 1996.

[MK98] J.F.A. Magarey and N.G. Kingsbury. Motion estimation using a complex-valued wavelet transform. *IEEE Transactions on Signal Processing*, 46(4) :1069–1084, April 1998.

[MLMS97] F. Mujica, J.-P. Leduc, R. Murenzi, and M.J.T. Smith. Spatio-temporal continuous wavelets applied to missile warhead detection and tracking. In J.Biemond and eds. E.J.Delp, editors, *SPIE-VCIP 3024*, pages 787–798, Bellingham, WA, 1997.

[MLMS00] F. Mujica, J.P. Leduc, R. Murenzi, and M. Smith. A new motion parameter estimation algorithm based on the continuous wavelet transform. *IEEE Trans.Image Process.*, 9 :873–888, May 2000.

[Mor98] F. Morier. *Méthodes de Représentation Hiérarchique du Contenu des Séquences d'Images Animées*. PhD thesis, Ireste, Nantes, septembre 1998.

[MP98a] E. Memin and P. Perez. Dense estimation and object-based segmentation of the optical flow with robust techniques. *IEEE Transactions on Image Processing*, 7(5) :703–719, May 1998.

[MP98b] E. Memin and P. Perez. A multigrid approach for hierarchical motion estimation. *Proceedings of International Conference on Computer Vision*, pages 933–938, 1998.

[MSB97a] J. Mendelsohn, E. Simoncelli, and R. Bajcsy. Discrete-time rigidity-constrained optical flow. In *Seventh International Conference on Computer Analysis of Images and Patterns*, Kiel, Germany, September 1997.

[MSB97b] J. Mendelsohn, E. Simoncelli, and R. Bajcsy. Discrete-time rigidity-constrained optical flow assuming planar structure. Technical report, GRASP laboratory technical report, February 1997.

[MWC02] D. Marpe, T. Wiegand, and H.L. Cycon. Design of a highly efficient wavelet-based video coding scheme. In *Proceedings of SPIE Visual Communications & Image Processing (VCIP)*, volume 4671, San Jose, USA, January 2002.

[MZ92] S.G. Mallat and S. Zhong. Characterization of signals from multiscale edges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(7), July 1992.

[OB98] J.M. Odobez and P. Bouthemy. Direct incremental model-based image motion segmentation for video analysis. *Signal Processing*, 66(2) :143–155, 1998.

[OMML98] L. Oisel, E. Mémin, L. Morin, and C. Labit. Epipolar constrained motion estimation for reconstruction from video sequences. In *Spie Conf. on Visual Communications and Image Processing, VCIP*, volume 3309, San-Jose, California, January 1998.

[PF94] T. Papadopoulo and O. Faugeras. Analyse du mouvement tridimensionnel à partir de séquences d'images en utilisant des surfaces Rapport n 2167, programme 4 (robotique, image et vision, projet robotvis), INRIA, January 1994.

[PK00] H.W. Park and H.S. Kim. Motion estimation using low-band shift method for wavelet-based moving picture coding. *IEEE transactions on image processing*, April 2000.

[PPB01] B. Pesquet-Popescu and V. Bottreau. Three-dimensional lifting schemes for motion compensated video compression. In *Proc. IEEE ICASSP*, Salt Lake City, 7-11 mai 2001.

[Que01] G.M. Quenot. Computation of optical flow using dynamic programming. Technical report, LIMSI-CNRS, Université Paris-sud, Orsay, 2001.

[Ric03] Iain E.G. Richardson. *H.264 and MPEG-4 Video Compression*. Wiley, 2003.

[SCD01] J.L. Starck, E. Candes, and D.L. Donoho. Image restoration by the curvelet transform. In *Proceedings of ICISP, Int'l Conference on Image and Signal Processing*, Agadir, Maroc, 2001.

[Sha93] J.M. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. *IEEETSP*, 41(12) :3445–3462, 1993.

[She92] M.J. Shensa. The discrete wavelet transform : Wedding the à trous and mallat algorithm. *IEEE Transactions on signal processing*, 40(10) :2464–2482, October 1992.

[Sim98] E. P. Simoncelli. *Handbook of computer vision and applications*, chapter Bayesian multiscale differential optical flow. Academic Press, 1998.

[SMB00] J.L. Starck, F. Murtagh, and A. Bijaoui. *Image Processing and Data Analysis ; the multiscale approach*. Cambridge University Press, 1998, reprinted 2000.

[SP96] A. Said and W.A. Pearlman. A new fast and efficient image codec based on set partitioning in hierarchical trees. *IEEE Transactions on Circuits and Systems for Video Technology*, 6 :243–250, June 1996.

[SR00] O. Sukmarg and K. R. Rao. Fast object detection and segmentation in MPEG compressed domain. In *Proceedings of IEEE TENCON 2000*, Kuala Lumpur, Malaysia, September 2000.

[SRT93] M. Shah, K. Rangarajan, and P.S. Tsai. Motion trajectories. *IEEE Transactions on Systems, Man and Cybernetics*, 23(4) :1138–1150, July 1993.

[ST01] A. Secker and D. Taubman. Motion-compensated highly scalable video compression using an adaptative 3d wavelet transform based on lifting. *IEEE*, 2001.

[Swe95] W. Sweldens. *The Construction and Application of Wavelets in Numerical Analysis*. PhD thesis, University of South Carolina, May 1995.

[Tau00] D. Taubman. High performance scalable image compression with ebcot. *IEEE Transactions on Image Processing*, 9(7) :1158–1170, July 2000.

[TL94] G. Tziritas and C. Labit. *Motion Analysis for Image Sequence Coding*, volume 4 of *Advances in Image Communication*. Elsevier Science, 1994.

[Tor95] B. Torresani. *Analyse continue par ondelettes*. Savoirs actuels. CNRS Editions, 1995.

[Tru98] F. Truchetet. *Ondelettes pour le signal numérique*. Hermes, 1998.

[TZ94] D. Taubman and A. Zakhor. Multirate 3-d subband coding of video. *IEEE Transactions on Image Processing*, 3(5) :572–588, september 1994.

[Van98] P. Vandergheynst. *Directional Wavelets and Wavelets on the Sphere*. PhD thesis, Université Catholique de Louvain-la-neuve, 1998.

[VG03] J Vieron and C. Guillemot. Low rate fgs video compression based on motion-compensated spatio-temporal wavelet analysis. In *Proc. of the SPIE Intl Conference on Visual Communication and Image Processing VCIP'03*, 2003.

[VMK95] Dang V.N., A.R. Mansouri, and J. Konrad. Motion estimation for region-nased video coding. In *Proceedings of ICIP*, Washington, D.C., 1995.

[VPZ98] J. Vass, K. Palaniappan, and X. Zhuang. Automatic spatio-temporal video sequence segmentation. In *Proceedings of ICIP, IEEE International Conference on Image Processing*, pages 958–962, Chicago, IL, October 1998.

[WA94a] J.Y.A. Wang and E.H. Adelson. Spatio-temporal segmentation of video data. In *Proceedings of SPIE on Image and Video Processing II, 2182*, pages 120–131, San Jose, February 1994.

[WA94b] Y. Weiss and E.H. Adelson. Perceptually organized em : A framework for motion segmentation that combines information about form and motion. Technical Report 315, MIT Media Lab Perceptual Computing Section TR, 1994.

[Wen04] S. Wenger. A database of video sequences from the tml project. http ://www.stewe.org/vceg.org/sequences.htm, 2004.

[Wis99] L. Wiskott. Segmentation from motion : Combining gabor- and mallat-wavelets to overcome the aperture and correspondence problems. *Pattern Recognition*, 32(10) :1751–1766, 1999.

[WKCL98] Y.T. Wu, T. Kanade, J. Cohn, and C.C. Li. Optical flow estimation using wavelet motion model. In *Sixth International Conference on Computer Vision*, pages 992–998. Narosa Publishing House, 1998.

[WL01] J.W. Woods and G. Lilienfeld. A resolution and frame-rate scalable subband/wavelet video coder. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(9) :1035–1044, September 2001.

[WM95] J. Weber and J. Malik. Robust computation of optical flow in a multi-scale differential framework. *Computer Vision*, 14 :67–81, 1995.

[WXCM99] A. Wang, Z. Xiong, P.A. Chou, and S. Mehrotra. Three-dimensional wavelet coding of video with global motion compensation. In *DCC '99 : Proceedings of the Conference on Data Compression*, page 404. IEEE Computer Society, 1999.

[XIP04] XIPH. videos database. http ://media.xiph.org/video/derf/, 2004.

[XLZX00] J. Xu, S. Li, Y.Q. Zhang, and Z. Xiong. A wavelet video coder using three dimensional embedded subband coding with optimized truncation (3-d escot). *Proceedings of IEEE Pacific-Rim Conference on Multimedia (PCM)*, Dec. 2000.

Bibliographie partie II

[AAE+95] A. Arneodo, F. Argoul, Bacry E., Elezgaray J., and Muzy J.F. *Ondelettes, multifractales et turbulences ; de l'ADN aux croissances cristallines*. Diderot, Sciences en actes, 1995.

[AH98] S. Aksoy and R.M. Haralick. Content-based access of image and video libraries. In *Proceedings. IEEE Workshop on textural features for image database retrieval*, pages 45–49, 21 June 1998.

[Bha67] C.G. Bhattacharya. A simple method of resolution of resolution of a distribution into gaussian components. *Biometrics*, 23 :115–135, 1967.

[BMD04a] P. Brault and A. Mohammad-Djafari. Bayesian segmentation of video sequences using a Markov-Potts model. *WSEAS Transactions on Mathematics*, 3(1) :276–282, January 2004.

[BMD04b] P. Brault and A. Mohammad-Djafari. Bayesian wavelet domain segmentation. In *Proceedings of the AIP, American Institute of Physics, for the International Workshop, MaxEnt, on Bayesian Inference and Maximum Entropy Methods*, pages 19–26, MaxPlanck Institute für Statistics, Garching, Germany, July 2004.

[BMD05a] P. Brault and A. Mohammad-Djafari. Segmentation bayesienne dans le domaine ondelettes (poster). In *Colloque Alain BOUYSSY (sans actes)*, Université Orsay Paris-sud, Février 2005. (poster présenté pour le Laboratoire des Signaux et Systèmes CNRS UMR8506/Supélec).

[BMD05b] P. Brault and A. Mohammad-Djafari. Unsupervised bayesian wavelet domain segmentation using a potts-markov random field modeling. *Journal of Electronic Imaging*, January 2005. (accepted for publication).

[Boc04] H-H. Bock. Clustering methods - a review of classical and recent approaches. In *Proceedings of Modelling, Computation and Optimization in Information Systems and Management Sciences*, Metz, France, July 2004.

[BS92] C.A. Bouman and M. Shapiro. Multispectral image segmentation using a multiscale model. *ICASSP, IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 565–568, March 1992.

[BS94] C.A. Bouman and M. Shapiro. A multiscale random field model for Bayesian image segmentation. *IEEE Transactions on Image Processing*, 3(2) :162–177, March 1994.

[CA79] G.B. Coleman and H.C. Andrews. Image segmentation by clustering. In *Proceedings of IEEE*, volume 67, pages 773–785, 1979.

[Can86] J.F. Canny. A computational approach to edge detection. *IEEE Transactions on PAMI*, 8(6) :679–698, 1986.

[CB97] M.S. Crouse and R.G. Baraniuk. Contextual hidden Markov models for wavelet-domain signal processing. In *Proc. of the 31th Asilomar Conf. on Signals, Systems, and Computers*, volume 1, pages 95–100, Pacific Grove, CA., November 1997.

[CB99] H. Choi and R.G. Baraniuk. Image segmentation using wavelet-domain classification. In *SPIE Conference Proceedings on Mathematical Modeling, Bayesian Estimation and Image Processing*, Denver, July 1999.

[CB01a] H. Cheng and C.A. Bouman. Multiscale Bayesian segmentation using a trainable context model. *IEEE Transactions on Image Processing*, 10(4) :51–525, 2001.

[CB01b] H. Choi and R.G. Baraniuk. Multiscale image segmentation using wavelet-domain hidden Markov models. *IEEE Transactions on Image Processing*, 10(9) :1309–1321, September 2001.

[CH80] R. Conners and C. Harlow. A theoretical comparison of texture algorithms. *IEEE Transactions on PAMI*, 2(3) :204–222, 1980.

[CNB98] M.S. Crouse, R.D. Nowak, and R.G. Baraniuk. Wavelet-based statistical signal processing using hidden Markov models. *IEEE Transactions on Signal Processing*, 46(4) :886–902, April 1998.

[CP95] J.P. Cocquerez and S. Philipp. *Analyse d'Images : Filtrage et Segmentation*. Masson, 1995.

[Dem02] G. Demoment. Probabilités : modélisation des incertitudes, inférence logique, et traitement des données expérimentales. Première partie : bases de la théorie. Cours de l'Université de Paris-Sud, Faculté des Sciences d'Orsay, Groupe des Problèmes Inverses, Laboratoire des Signaux et Systèmes, Supélec, 2002.

[Dem04] G. Demoment. Outils mathematiques pour le traitement du signal (résumé de cours). Ist-m2r-sseti : Master en information, systèmes et technologie de l'Université de Paris-Sud, Centre d'Orsay (formation à la recherche en sciences des systèmes embarqués et traitement de l'information), Groupe des Problèmes Inverses, Laboratoire des Signaux et Systèmes, Supélec, 2004.

[Der87] R. Deriche. Using Canny's criteria to derive a recursively implemented optimal edge detector. *International Journal of Computer Vision*, 1(2) :167–187, 1987.

[FMD03] O. Féron and A. Mohammad-Djafari. A hidden Markov model for Bayesian data fusion of multivariate signals. In *Proceedings of Fifth International Triennal Calcutta Symposium on Probability and Statistics*, Dept. of Statistics, Calcutta University, India, December 2003.

[FMD05] O. Féron and A. Mohammad-Djafari. Image fusion and unsupervised joint segmentation using a HMM and MCMC algorithms. *Journal of Electronic Imaging*, 14(2), june 2005.

[FX01] G. Fan and X.G. Xia. A joint multicontext and multiscale approach to Bayesian image segmentation. *IEEE Transactions on Geoscience and Remote Sensing*, 39(12) :2680–2688, December 2001.

[FX02] G. Fan and X.G. Xia. Wavelet-based texture analysis and synthesis using hidden Markov models. *IEEE Transactions on Geoscience and Remote Sensing*, 40(1) :229–229, January 2002.

[GG84] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of image. *IEEE PAMI*, 6(6) :721–741, November 1984.

[GG00] P. Gerard and A. Gagalowicz. Three-dimensional model-based tracking using texture learning and matching. *Pattern Recognition Letters*, 21(13–14) :1095–1103, December 2000.

[GGG87] S. Geman, D. Geman, and C. Graffigne. Locating texture and object boundaries. In Ed. P.A. Devijver and J.Kittler, editors, *Pattern Recognition Theory and Application*, Heidelberg, 1987. Springer-Verlag.

[Har84] R.M. Haralick. Digital step edges from zero-crossings of second directional derivative. *IEEE Trans. on PAMI*, 6(1) :58–68, 1984.

[Haz00] G.G. Hazel. Multivariate Gaussian MRF for multispectral scene segmentation and anomaly detection. *IEEE Transactions on Geoscience end Remote Sensing*, 38(3) :1199 – 1211, May 2000.

[HMD05] F. Humblot and A. Mohammad-Djafari. Super-resolution using hidden markov model and bayesian detection estimation framework. special issue on super-resolution imaging : Analysis, algorithms, and applications. In *EURASIP, Journal on Applied Signal Processing (accepted, planned to be published end of 2005)*, 2005.

[HSD73] R.M. Haralick, K. Shanumugam, and I. Dinstein. Texture features for image classification. *IEEE Transactions on Systems, Man and Cybernetics*, 3(6) :610–621, November 1973.

[HZ04] A. Hafiane and B. Zavidovique. Automating GIS image retrieval based on MCM. In *ICIAR*, pages 787–794, 2004.

[Idi01] J. (Ed.) Idier. *Approche bayésienne pour les problemes inverses*. Hermes Science, 2001.

[IMD04] M. Ichir and A. Mohammad-Djafari. Bayesian based source separation for nonstationary positive sources. In *Bayesian Inference and Maximum Entropy Methods*, Munich, Germany, 2004. MaxEnt Workshops. Maxent04.

[Jai89] A.K. Jain. *Fundamental of Digital Image Processing*. Prentice-Hall, 1989.

[Jay95] E.T. Jaynes. *Probability Theory : The Logic of Science*. 1995.

[JB83] B Julesz and Bergen. Textons. *Bell Systems Technical Journal*, 1883.

[JD88] A.K. Jain and R.C. Dubes. *Algorithms for Clustering Data*. Prentice-Hall, New-Jersey, 1988.

[JDBA00] S. Jehan, E. Debreuve, M. Barlaud, and G. Aubert. Segmentation spatio-temporelle d'objets en mouvement dans une séquence vidéo par contours actifs déformables. In *Proceedings of RFIA*, Paris, Fevrier 2000.

[JPZ88] M. Jourlin, J.C. Pinoli, and R. Zeboudj. Contrast definition and contour detection for logarithmic images. *Journal of Microscopy*, 156 :33–40, 1988.

[Jul83] B Julesz. Visual pattern discrimination. *IRE (IEEE) Transaction on Information Theory*, 8(1) :84–92, February 1883.

[KC89] J.M. Keller and S. Chen. Texture description and segmentation through fractal geometry. *Computer Vision, Graphics and Image Processing*, 45 :150–166, 1989.

[KLM04] T. Kunlin, L. Lacassagne, and A. Merigot. A fast image segmentation scheme. In *Proceedings of ICIP04 International Conference on Image Processing*, 2004.

[Mac00] D.J.C. MacKay. *Information Theory, Inference and Learning Algorithms*. Cambridge, 2000.

[Mal01] S.G. Mallat. *A Wavelet Tour of Signal Processing*. Addison Westley, 1 and 2 edition, 1997 and 2001.

[MD01] A. Mohammad-Djafari. *Detection-Estimation ; Graduated Course, Department of Electrical Engineering.* University of Notre-Dame, IN, USA, 2001. djafari@lss.supelec.fr.

[MD04] A. Mohammad-Djafari. *Cours de DEA ATS : Problèmes Inverses*. LSS, Laboratory of Signals and Systems, Supélec, Gif sur Yvette, France., 2004. djafari@lss.supelec.fr.

[MFMD04] A. Mohammadpour, O. Féron, and A. Mohammad-Djafari. Bayesian segmentation of hyperspectral images. In *MaxEnt04*, 2004.

[MH92] S. G. Mallat and W. L. Hwang. Singularity detection and processing with wavelets. *IEEE Transactions on Information Theory*, 38(2) :617–643, 1992.

[MZ92] S. Mallat and S. Zhong. Characterization of signals from multiscale edges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(7), July 1992.

[Pen84] A.P. Pentland. Fractal-based description of natural scenes. *IEEE Transactions on PAMI*, pages 661–674, 1984.

[PKLH96] J. Pesquet, H. Krim, H. Leporini, and E. Hamman. Bayesian approach to the best basis selection. In *ICASSP*, Atlanta, May 1996.

[PS85] F.G. Peet and T.S. Sahota. Surface curvature as a measure of image texture. *IEEE Transactions on PAMI*, 7(6) :734–738, November 1985.

[PSWS03] J. Portilla, V. Strela, W. Wainwright, and E.P. Simoncelli. Image denoising using scale mixtures of Gaussians in the wavelet domain. *IEEE Trans. on Image Processing*, 12(11), 2003.

[Rao93] Lohre Rao. Towards a texture naming system. *Proceedings of the IEEE fourth Conference on Visualization*, 1993.

[RCB01] J.K. Romberg, H. Choi, and R.G. Baraniuk. Bayesian tree-structured image modeling using wavelet-domain hidden Markov models. *IEEE Transactions on Image Processing*, 10(7) :1056 – 1068, July 2001.

[Rob92] C. Robert. *L'Analyse Statistique Bayesienne*. Economica, Paris, 1992.

[Rob96] C. Robert. *Méthodes de Monte Carlo par chaines de Markov*. Economica, Paris, 1996.

[SCZ89] T. Simchony, R. Chellapa, and Lichtenstein Z. The graduated non-convexity algorithm for image estimation using compound Gauss-Markov field models. In *Proc. ICASSP89*, Glasgow, May 1989.

[SF02] X. Song and G. Fan. A study of supervised, semi-supervised and unsupervised multiscale Bayesian image segmentation. In *MWSCAS02, 45th Midwest Symposium on Circuits and Systems*, volume 2, pages 371–374, 2002.

[SF03a] X. Song and G. Fan. Unsupervised Bayesian image segmentation using wavelet-domain hidden Markov models. In *Proc. of ICIP, International Conference on Image Processin*, volume 2, pages 423–426, September 2003.

[SF03b] X. Song and G. Fan. Unsupervised image segmentation using wavelet-domain hidden Markov models. In *Proceedings of SPIE Wavelets X in applications in signal and image processing*, San Diego, 2003.

[SF04] X. Song and G. Fan. Unsupervised image segmentation by exploiting likelihood disparity of texture behavior. *Submitted to IEEE Transactions on Image Processing*, pages 1–30, April (submission date) 2004.

[Sha93] J. M. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. *IEEETSP*, 41(12) :3445–3462, 1993.

[SHB99] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis and Machine Vision*. PWS publishing, second edition, 1999.

[SIP] SIPI. Images and videos database. http ://sipi.usc.edu/publications.html.

[SMD02] H. Snoussi and A. Mohammad-Djafari. Fast joint separation and segmentation of mixed images. *Journal of Electronic Imaging*, 13(2) :349–361, April 2002.

[Vos86] R.F. Voss. Characterization and measurement of random fractals. *Physica Scripta*, 13 :27, March 1986.

[WP94] E. Walter and L. Pronzato. *Identification de Modèles Paramétriques, à partir de données expérimentales*. Modélisation Analyse Simulation Commande. Masson, 1994.

[Wu82] F.Y. Wu. The Potts model. *Review of Modern Physics*, 54(1) :235–268, January 1982.

[ZC93] J. Zerubia and R. Chellapa. Mean field annealing using compound Gauss-Markov random fields for edge detection and image estimation. *IEEE Trans. on Neural Networks*, 4 :703–709, 1993.

Bibliographie personnelle

Les références [2, 3, 4, 5, 6, 7, 8, 9] concernent directement cette thèse. Les autres références ont été publiées durant ou avant cette thèse sur d'autres sujets.

[1] http ://braultp.free.fr.

[2] P. Brault and A. Mohammad-Djafari. Unsupervised bayesian wavelet domain segmentation using a potts-markov random field modeling. *Journal of Electronic Imaging*, January 2005. (accepted).

[3] P. Brault and A. Mohammad-Djafari. Segmentation bayesienne dans le domaine ondelettes (poster). In *Colloque Alain BOUYSSY (sans actes)*, Université Orsay Paris-sud, Février 2005. (poster présenté pour le Laboratoire des Signaux et Systèmes CNRS UMR8506/Supélec).

[4] P. Brault and A. Mohammad-Djafari. Bayesian wavelet domain segmentation. In *Proceedings of the AIP, American Institute of Physics, for the International Workshop, MaxEnt, on Bayesian Inference and Maximum Entropy Methods*, pages 19–26, MaxPlanck Institute für Statistics, Garching, Germany, July 2004.

[5] P. Brault and A. Mohammad-Djafari. Bayesian segmentation of video sequences using a Markov-Potts model. *WSEAS Transactions on Mathematics*, 3(1) :276–282, January 2004.

[6] P. Brault. On the performances and improvements of motion-tuned wavelets for motion estimation. *WSEAS Transactions on Electronics*, 1(1) :174–180, 2004.

[7] P. Brault. A new scheme for object-oriented video compression and scene analysis, based on motion tuned spatio-temporal wavelet family and trajectory identification. In *Proceedings of the 3rd IEEE International Symposium on Signal Processing and Information Technology*, Darmstadt, 2003.

[8] P. Brault and M. Vasiliu. Motion-compensated spatio-temporal filtering with wavelets. *WSEAS Transactions on Computers*, 2(4) :1131–1140, 2003.

[9] P. Brault. Motion estimation and video compression with spatio-temporal motion-tuned wavelets. *WSEAS Transactions on Mathematics*, 2(1 & 2) :67–78, 2003.

[10] P. Brault, J.L. Starck, and P. Beauvillain. Characterization of nanostructures stm images with the wavelet and ridgelet transforms. In *Proceedings of the IAPR-ICISP, International Conference on Image and Signal Processing*, volume 1, pages 232–241, Agadir, 2003.

[11] P. Brault and H. Mounier. Automated, transformation invariant, shape recognition through wavelet multiresolution. In *Proceedings of the SPIE, International Society for Optical En-*

gineering Wavelets : Applications in Signal and Image Processing IX, Andrew F. Laine ; Michael A. Unser ; Akram Aldroubi ; Eds., volume 4478, pages 434–443, San Diego, 2001.

[12] P. Brault and H. Mounier. Wavelet multi-resolution transform applied to shape recognition based on a curvature criterion. In *Proceedings of the IAPR International Conference on Image and Signal Processing*, volume 1, pages 250–258, Agadir, 2001.

[13] P. Brault, H. Mounier, N. Petit, and P. Rouchon. Flatness based tracking control of a manoeuvrable vehicle : the $\pi - car$. In *Proceedings of the MTNS, International Conference on Mathematical Theory of Networks and Systems*, pages 1–7, Perpignan, 2000.

[14] C. Goutelard and P. Brault. Fractal approach of the ionospheric channel scattering function. In *Proceedings of the COST 257 Congress*, El Arenosillo, 1999.

[15] P. Brault. Engineer thesis : Fractal analysis of the random propagation channels in the ionosphere. Technical report, PARIS, 1998.

[16] P. Brault. Digital phase-locked loops. engineer cycle probing oral. dir. M. Bellanger. Technical report, CNAM Paris, 1996.

Index

Résumé
La première partie de ce mémoire présente une nouvelle vision de l'estimation de mouvement, et donc de la compression, dans les séquences vidéo. D'une part, nous avons choisi d'aborder l'estimation de mouvement à partir de familles d'ondelettes redondantes adaptées à différentes transformations, dont, plus particulièrement, la vitesse. Ces familles, très peu connues, ont déjà été étudiées dans le cadre de la poursuite de cibles. D'autre part, les standards de compression actuels comme MPEG4 prennent en compte une compression objet mais ne calculent toujours que de simples vecteurs de mouvements de "blocs". Il nous a paru intéressant de chercher à mettre en oeuvre ces familles d'ondelettes car 1) elle sont construites pour le calcul de paramètres sur plusieurs types de mouvement (rotation, vitesse, accélération) et 2) nous pensons qu'une approche de l'estimation basée sur l'identification de trajectoires d'objets dans une scène est une solution intéressante pour les méthodes futures de compression. En effet nous pensons que l'analyse et la compréhension des mouvements dans une scène est une voie pour des méthodes de compression "contextuelles" performantes.

La seconde partie présente deux développements concernant la segmentation non-supervisée dans une approche bayésienne. Le premier, destiné à réduire les temps de calcul dans la segmentation de séquences vidéo, est basé sur une mise en oeuvre itérative, simple, de la segmentation. Il nous a aussi permis de mettre une estimation de mouvement basée sur une segmentation "région" (voire objet). Le second est destiné à diminuer les temps de segmentation d'images fixes en réalisant la segmentation dans le domaine des ondelettes. Ces deux développements sont basés sur une approche par estimation bayésienne utilisant un modèle de champ aléatoire de Potts-Markov (PMRF) pour les étiquettes des pixels, dans le domaine direct, et pour les coefficients d'ondelettes. Il utilise aussi un algorithme itératif de type MCMC (Markov Chain Monte Carlo) avec échantillonneur de Gibbs. L'approche initiale, directe, utilise un modèle de Potts avec voisinage d'ordre un. Nous avons développé le modèle de Potts pour l'adapter à des voisinages convenant aux orientations privilégiées des sous-bandes d'ondelettes. Ces réalisations apportent, à notre connaissance, des approches nouvelles dans les méthodes de segmentation non-supervisées.

Abstract
The first part of this thesis presents a new vision of the motion estimation problem, and hence of the compression of video sequences. On one hand, we have chosen to investigate motion estimation from redundant wavelet families tuned to different kind of transformations and, in particular, to speed. These families, not well known, have already been studied in the framework of target tracking. On the other hand, today video standards, like MPEG4, are supposed to realize the compression in an object-based approach, but still compute raw motion vectors on "blocks". We thus implemented these wavelet families because 1) they are built to perform motion parameter quantization on several kinds of motions (rotation, speed, acceleration) and 2) we think that an approach of motion estimation based on the trajectory identification of objects motions in a scene is an interesting solution for future compression methods. We are convinced that motion analysis, and understanding, is a way of reaching powerful "contextual" compression methods.

The second part introduces two new methods and algorithms of unsupervised classification and segmentation in a Bayesian approach. The first one, dedicated to the reduction of computation times in the segmentation of video sequences, is based on an iterative, simple, implementation of the segmentation. It also enabled us to set up a motion estimation based on objects segmentation. The second is aimed at reducing the segmentation times, for images, by performing the segmentation in the wavelet domain. Both algorithms are based on a Bayesian estimation approach with a Potts-Markov random field (PMRF) model for the labels of the pixels, in the direct domain, and for the wavelet coefficients. It also uses an iterative MCMC (Markov Chain Monte-Carlo) algorithm based on a Gibbs sampler. The initial PMRF model, in the direct domain, works with a first order neighboring. We have developed the PMRF model to tune it to the priviledged orientations of the wavelet subbands. These realizations provide, to our knowledge, new approaches to unsupervised segmentation methods.